DRIVVEN

User's Manual
Version 3.110
October 2011

Drivven, Inc. • 12001 Network Blvd, Bldg E, Suite 110 • San Antonio, Texas 78249 • USA
Phone : 210.248.9308
Web : www.drivven.com , E-mail : info@drivven.com

Table of Contents

Introduction 4
Installing and Activating CalVIEW 6
CalVIEW Palettes....9
Block Diagram Palette 9
Front Panel Palette....14
CalVIEW Target 16
Basic Framework....16
CalVIEW Master VI....16
Specifying the Calibration File 22
CalPoints 24
Reserved CalPoints....24
CalPoint Configuration Dialogs....25
SGL, I32, I8, U32, U8 CalPoints 26
Array CalPoints 26
String CalPoints 27
Boolean CalPoints 27
Lookup Table CalPoints....28
Enumerated CalPoints 29
Adapt Points 30
Fault Points....31
VxWorks Setup....33
Building Startup Executable Applications with CalVIEW 33
Backup and Restore of a Real-Time System....33
CalVIEW Host 36
Lookup Table Controls....37
1D Tables....38
2D Tables....41
CalScope....43
CalTrend....46
Faults....48
Sub-Hosts....48
Other Host VI Tools 49
CalVIEW Console 51
Menu Items....54
CalVIEW Controls....56
Mini Mode....57
CalVIEWConfig.ini 57
Saving and Loading CalVIEW settings 59
Finding a Target 62
Pairing CalPoints....64
Taking a Snapshot....65
CalDiff (Calibration File Difference) 66
Calibration File Manager....67
Unit Conversions 69
Unit Conversion Rules....71

Example of Unit Conversions 72
Simple Example 73
Basic Project Example 73
Frequently Asked Questions 75
Online Tools 78

Introduction

CalVIEW is a LabVIEW-based calibration interface toolkit for efficiently communicating and calibrating data parameters (CalPoints) and faults (Fault Points) between a LabVIEW target and a Windows host via Ethernet. It provides a user-friendly implementation for real-time target programming and an interface console for pairing RT target parameters to a LabVIEW Windows host VI. Furthermore, CalVIEW supports ASAM MCD-2 (A2L) file generation and communication with devices over ASAM MCD-3MC (ASAP3).

Features:

- Supported CalPoint data types:
  - SGL
  - I32, U32, I8, U8
  - 1D Lookup Table, 2D Lookup Table
  - 1D Array, 2D Array
  - String
  - Enumerated U32
  - Adapt Points for controls types
- Flexible Fault Point configuration and control
  - Up/Down counter to set/clear faults
  - Get and clear faults from the target or host
  - Sticky faults precise for a set time or key cycles
  - Color and sounds associated with fault severity
- ASAM MCD-3MC (ASAP3) Support
- Fast, efficient CalPoint communication
  - Up to 20Hz host display update rate (depends on target CPU load)
  - Target reports only requested visible CalPoints
  - Host reports only modified CalPoints
  - Hundreds of CalPoints can be embedded in target application
  - Allows deterministic loop execution on target
- Calibration management
  - CalPoint values can be saved to target as new default startup values
  - Target application does not require modification for new CalPoint defaults
  - Live loading of calibrations
  - Calibration file differencing tool included
  - Calibration file management tool
- Easy to setup and modify
  - Popup CalPoint Express VIs allow easy configuration and error checking without using target runtime resources
  - Programmer can add, modify and remove CalPoints without writing custom network communications code
  - CalPoints can exist in sub VIs without being wired up to the top level
  - CalPoints are graphically small and unobtrusive on the block diagram
- Multiple Host VIs and Sub-Host VIs
- Plotting and trending
  - CalScopes plot CalPoints at rates up to 1kHz
  - CalTrends plot CalPoints trends at rates up to 1Hz

  - User-configurable during runtime
  - Log displayed data to disk
- Unit conversion
  - User-specified unit conversions managed on host
  - User-specified default units and exceptions
- Drag-and-drop, keyboard, or automatic CalPoint pairing
- User interface can operate on Windows host using LabVIEW run-time engine

Installing and Activating CalVIEW

CalVIEW may be downloaded from the Drivven customer portal website at http://portal.drivven.com/SoftwareDownload. A password and a user name are required to access the site. Please contact Drivven for more details.

Recommended Minimum PC
CPU: 1.6GHz or faster (dual core recommended)
OS: Windows XP, Vista, or 7
RAM: 512MB or more
Hard Drive: 250MB or more free space
Host Software: LabVIEW Full 2011 (to create host applications)
Target Software: LabVIEW 2011 or LabVIEW Real-Time 2011

There are two use cases for CalVIEW. First, CalVIEW may be fully activated to allow CalPoints to be added or modified within LabVIEW RT applications. LabVIEW RT is required for this. Second, CalVIEW may be minimally activated for the use of the CalVIEW console on Windows. This allows users to monitor and calibrate pre-existing RT applications with embedded CalPoints, and only requires the free LabVIEW Runtime Engine. The user interface VI on the Windows host may be modified if LabVIEW Base (or higher) is installed. The Runtime Engine may be downloaded from the following address:

http://joule.ni.com/nidu/cds/view/p/id/2534/lang/en or search www.ni.com for “LabVIEW Run-Time Engine 2011”. The LabVIEW Runtime installer is ~209 Mb.

CalVIEW consists of three components; a LabVIEW RT VI palette (block diagram), LabVIEW Windows X-Control palette (front panel) and a CalVIEW console for Windows. The LabVIEW RT palette is used to instrument the LabVIEW RT application with CalPoints. The Windows X-Control palette provides specialty controls for interfacing with unique CalPoints on the target such as 1D and 2D lookup tables. The CalVIEW console is used to pair target CalPoints with LabVIEW Windows host VI controls/indicators and transfer the data to and from the host VI front panel.

CalVIEW uses several different network ports to communicate with the target. Table 1 describes the network connections used by CalVIEW. These ports should be unblocked by firewall software on the host computer.

Table 1: Network Connections

Connection Type	Port	Use
UDP	61570	CalVIEW Discovery
UDP	61571	CalVIEW Discovery
TCP	21	FTP File Transfer
TCP	22222	ASAP3 Communication
TCP	22255	Calibration Difference
TCP	32250	Host VI Communication
UDP	33505	CalVIEW Demon
UDP	33506	CalVIEW Demon
TCP	3363	VI Server

After installing CalVIEW, it must be activated in order to enable its components. Activation is initiated from the CalVIEW Windows console. Open the dialog found under the Help→Activate CalVIEW menu item (Figure 1). The Dialog box displays a Computer ID number which must be emailed to license@drivven.com along with your name, company name, address and telephone number. Please allow 1-3 business days to validate your account and generate a key. The Computer ID number uniquely identifies the computer being activated based on the MAC address(es) of your computer. An activation key or keys (32 Characters ex. 9DCJ-RN79-QBEE-TPKW-BBAB-GT3Z-RXYC-CFCW) will be generated by Drivven and sent to the customer via reply email. The key(s) will contain only the following characters A B C D E F G H J K L M N P Q R S T W X Y Z 3 4 7 9. The activation key(s) may expire at a specific time if configured to do so. The activation key(s) must then be entered into the Activate CalVIEW dialog box.

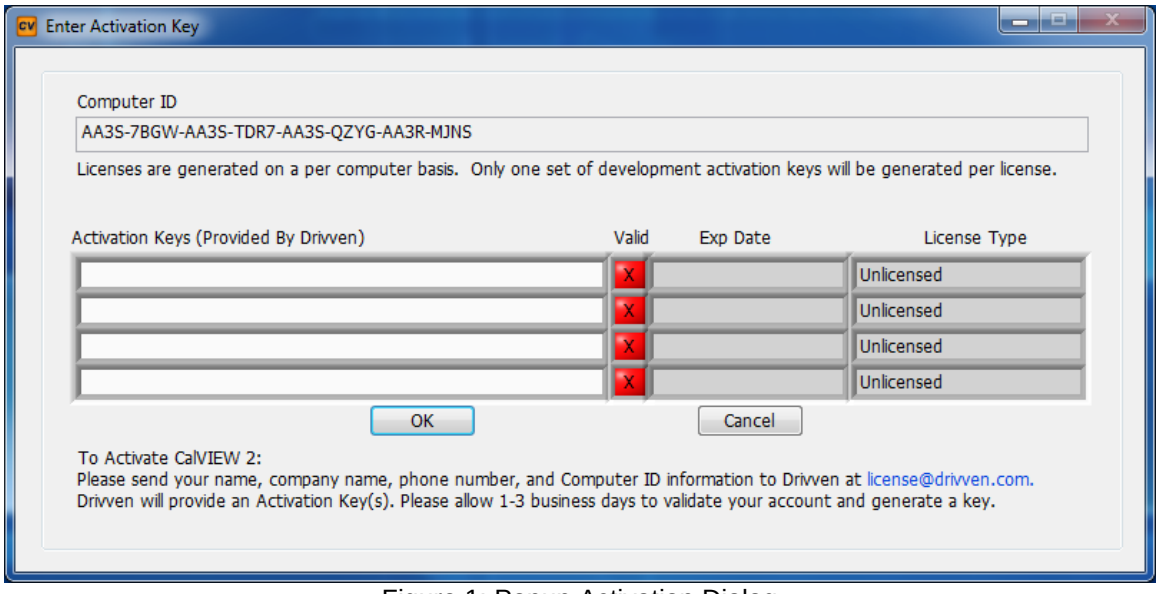

Enter Activation Key
Computer ID
AA3S-7BGW-AA3S-TDR7-AA3S-QZYG-AA3R-MJNS
Licenses are generated on a per computer basis. Only one set of development activation keys will be generated per license.
Activation Keys (Provided By Drivven)
Valid
Exp Date
License Type
Unlicensed
Unlicensed
Unlicensed
Unlicensed
OK
Cancel
To Activate CalVIEW 2:
Please send your name, company name, phone number, and Computer ID information to Drivven at license@drivven.com.
Drivven will provide an Activation Key(s). Please allow 1-3 business days to validate your account and generate a key.

Figure 1: Popup Activation Dialog

Computer ID: Unique computer identification number required to activate CalVIEW.

Activation Key: Drivven will provide this key via return email. Up to four separate activation keys may be entered.

Valid: Turns green when a valid key has been entered.

Exp Date: Shows the date and time when the license will expire or has expired.

License Type: Shows the license type.

OK: Accepts the changes made to the license information and closes the dialog box.

Cancel: Rejects the changes made to the license information and closes the dialog box.

CalVIEW Palettes

CalVIEW installs two types of palettes into LabVIEW the first is the block diagram palettes which contain all the subVIs required to create a CalVIEW target and add extra functionality to the host. The second palette is the front panel pallet which contains controls needed to add certain functionality to a CalVIEW host.

Block Diagram Palette

The CalVIEW LabVIEW palette (block diagram), found under Drivven→CalVIEW, consists of VIs that can be used to enhance CalVIEW functionality (Figure 2). The top left VI on the palette is used to manage the general functions of CalVIEW (CalVIEW Master). The rest of the VIs are separated into sub-palettes based on functionality. The target utilities palette contains a collection of VIs to be used in target applications and interface with the core CalVIEW functionality (Figure 3). The host utilities palette contains a collection of VIs designed to run in host VI and interface with the CalVIEW console (Figure 4). The CalPoints palette holds the different types of CalPoints (Figure 5). The adapt points palette may be used to modify control values from the target application while the target is running (Figure 6). Fault points are found on the fault points palette (Figure 7).

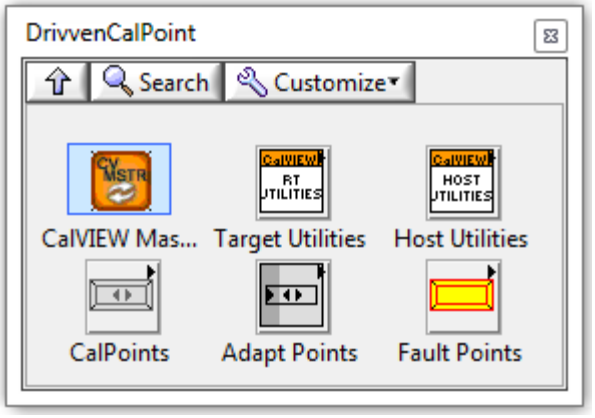


Figure 2: CalVIEW Palette

CalVIEWMaster Source.vi

The CalVIEW Master VI must be used once and only once in the top level RT VI. The CalVIEW Master VI is an express VI and serves two main functions. The first function is a collection and dialog display of all the CalPoints in the target application. This dialog runs on the host computer before the application is deployed to the target. It locates all of the CalPoints in an RT application and builds an internal table which is used during run-time. All of the CalPoints are checked for errors and any errors found are reported to the user within the dialog. The CalVIEW Master Configuration Dialog must be opened before

deploying the RT application if any CalPoints are added, removed or modified since last opening the dialog. Failure to open the configuration dialog after making CalPoint changes will result in undefined behavior. The configuration dialog box opens automatically when the user drops the CalVIEW Master VI or when the user double clicks on the VI.

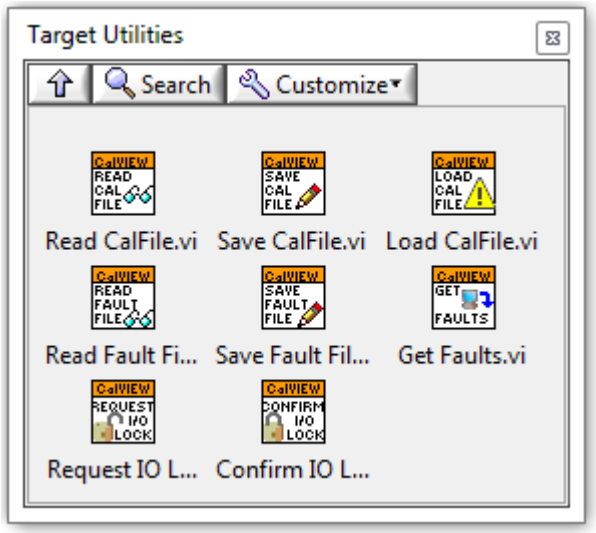
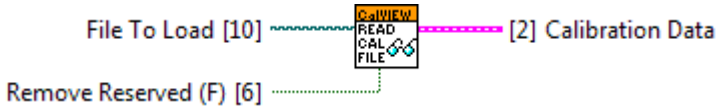


Figure 3: Target Utilities Palette

Read CalFile.vi
This VI reads the data from a calibration file. It may be used on the target to evaluate calibration data.



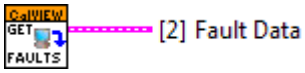
Save CalFile.vi
This VI saves a Calibration file in the default data directory. "Calibration File" should be set to the desired calibration file name without a file extension. Alternatively, "Calibration File" may be set to a plane text file (*.txt including the file extension) which points to the calibration file.
If "Controls Only" is set to TRUE, only CalPoint Controls will be saved in the calibration file. Otherwise, all CalPoints will be saved. This feature creates a smaller calibration file, especially when using large array indicators.
The "Saved File" indicator returns the final saved file name.

Load CalFile.vi
This VI loads a Calibration file from the default data directory while the target is running. The program should verify that the I/O is in a safe state before proceeding with loading a calibration file on a running target. Failure to lock the I/O properly could result in dangerous states. Live loading of calibration files must be enabled in CalVIEW Master and the confirm I/O Lock must be true.

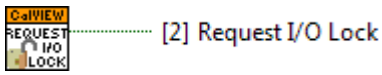
Read Fault File.vi
This VI reads the data from a fault file. It may be used on the target to evaluate fault data.

Save Fault File.vi
This VI saves a Fault file in the default data directory. "Fault File" should be set to the desired Fault file name without a file extension.
The "Saved File" indicator returns the final saved file name.

Get Faults.vi
This VI is used to get the current CalPoint Values. It should only be run on a running target.



Request IO Lock.vi
This VI reports the request for a hardware I/O lock so that a calibration file may be loaded without a restart. After the hardware I/O lock has been completed, use the confirm I/O Lock to proceed with loading the calibration file.



Confirm IO Lock.vi
This VI is used to confirm that the hardware I/O has been locked so that the calibration file can be loaded without restarting the application. The hardware I/O lock prevents undesired events during calibration file transition.

Warning: The user application must lock the hardware before setting the confirm hardware I/O lock. Failure to do so may result in undesired events.

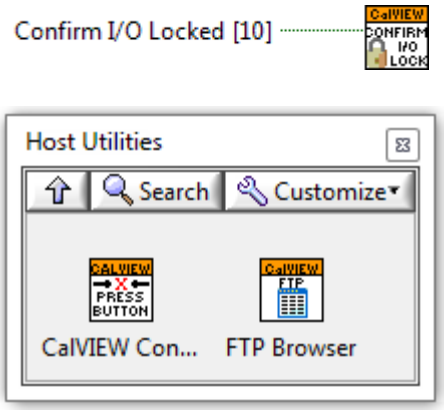
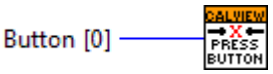
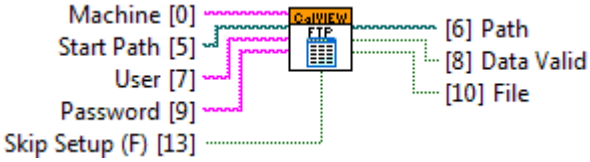


Figure 4: Host Utilities Palette

CalVIEW Console Button.vi
This VI is to be used in the host VI and "presses" some of the buttons on the CalVIEW console as if the user presses the button. This can be used to exit the user interface and CalVIEW console or save a calibration file.



FTP Browser.vi
This VI uses the FTP client to browse a remote file system and creates a local path for the remote machine to use. This VI should be used in the host application and not in the target application. The LabVIEW Internet Toolkit must be installed for this VI to work as a part of the LabVIEW development environment.



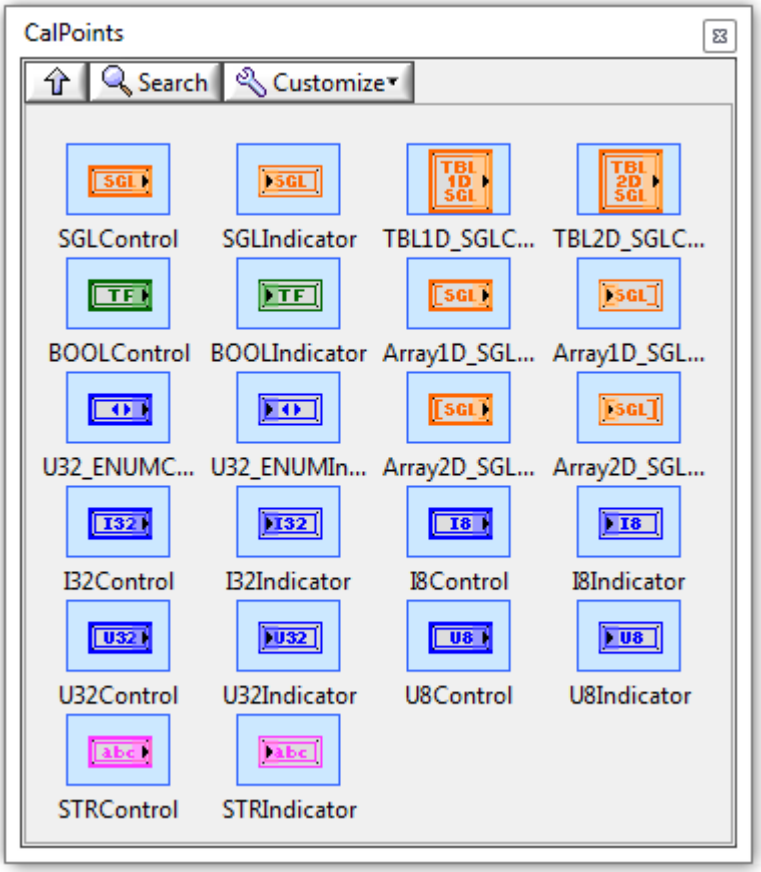


Figure 5: CalPoints Palette

Each type of CalPoint has a variety of parameters that are configured through a configuration dialog. Some types of CalPoints can programmatically adapt using Adapt Points. The Adapt Points must have a matching type and name to work properly.

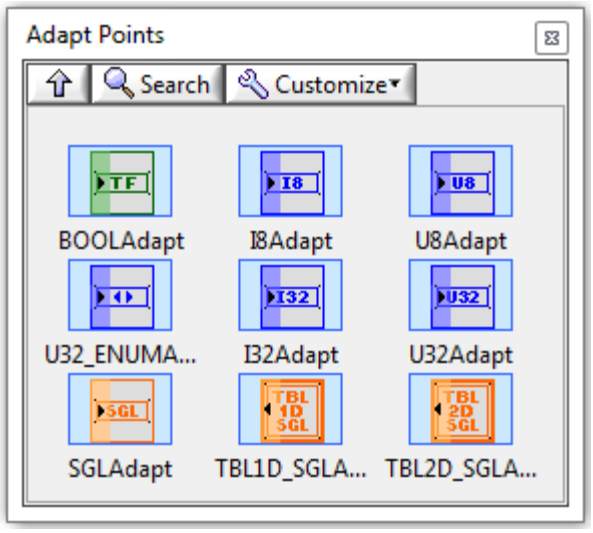


Figure 6: Adapt Points Palette

Fault points are used to manage fault conditions. The faults have 3 different target operations. Set, get, and clear which can be used to control the fault state.

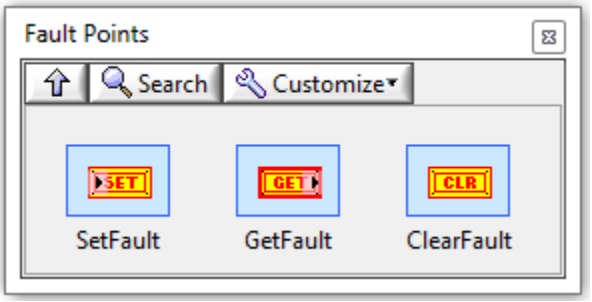


Figure 7: Fault Points Palette

Front Panel Palette

The CalVIEW LabVIEW palette (front panel), found under Drivven→CalVIEW, consists of controls that can be used to enhance CalVIEW functionality. (Figure 8) 1D and 2D tables pair with 1D and 2D table CalPoints. Bit fields pair with unsigned 32-bit integers. The path control pairs with strings and provides an ftp browser for remote targets. The text history control pairs with strings and keeps a history of the strings displayed. The rest of the controls do not pair with CalPoints. Instead, they connect to the CalVIEW console and provide special functionality. The sub-host controls allow the user to configure and open sub-hosts. The CalScope and CalTrend controls are only allowed once in a project. The faults control provides a connection to fault points and is only allowed once in a project.

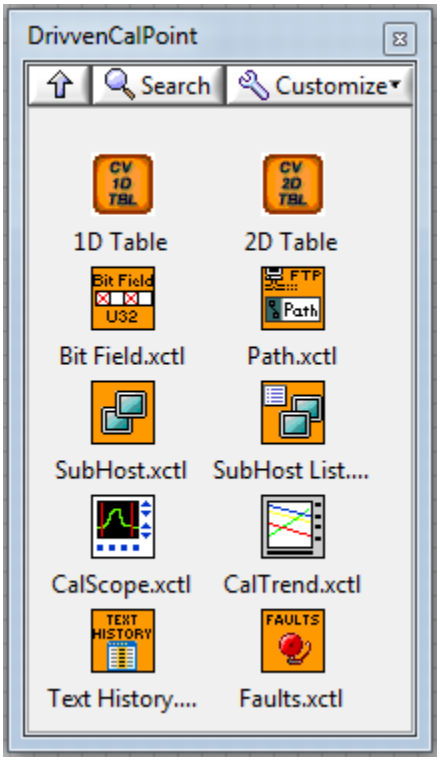


Figure 8: Controls Palette

CalVIEW Target

Basic Framework

A CalVIEW target VI must use a basic framework in order to work properly. (Figure 9) The framework starts with the CalVIEW Master VI controlling a case structure. The case structure true state then contains all the user code.

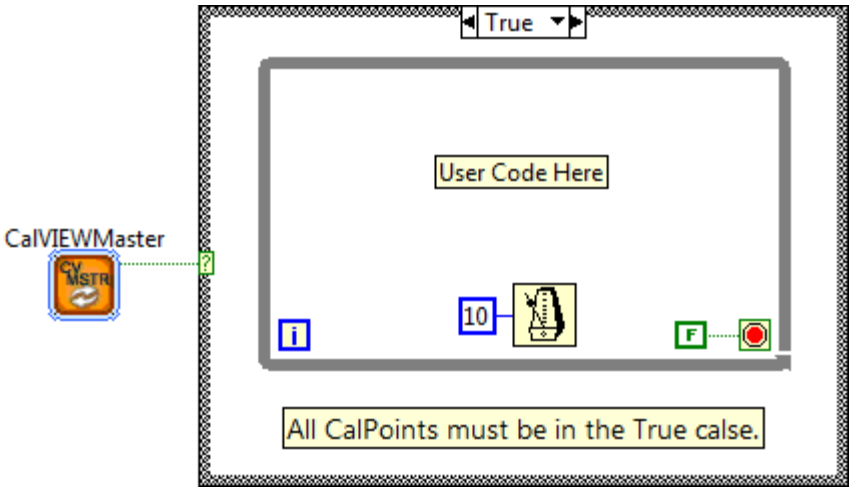


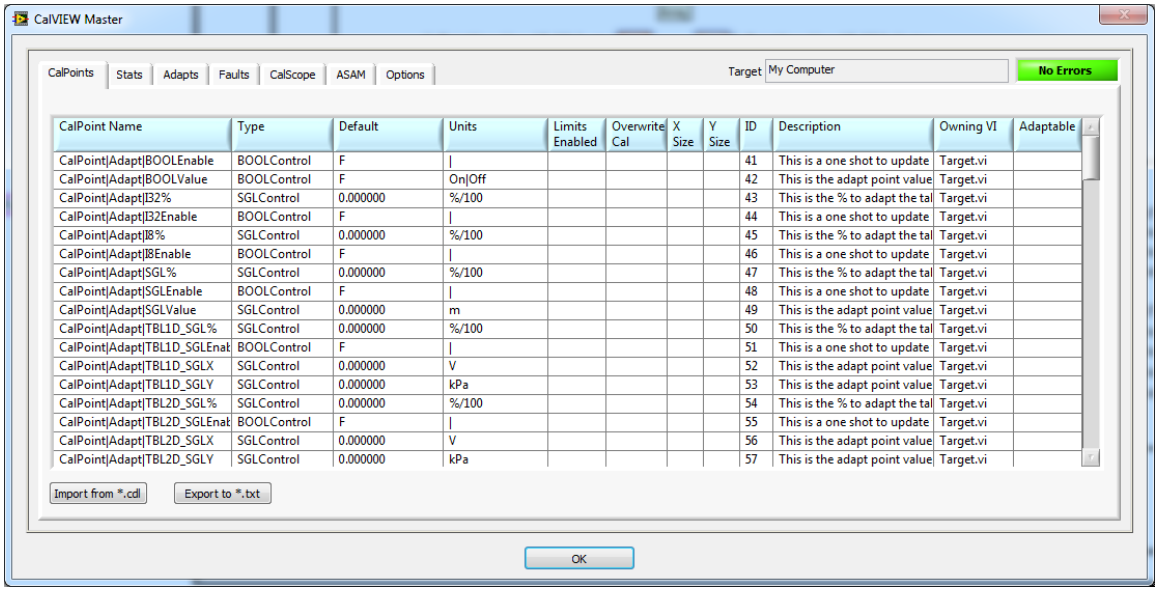
Figure 9: Basic Target Structure

CalVIEW Master VI

Figure 10: CalVIEW Master

The CalVIEW Master VI must be used once and only once in the top level target VI. The CalVIEW Master VI is an express VI and serves two main functions. The first function is a collection and dialog display of all the CalPoints, Adapt Point, and Fault Points in the target application. This dialog runs on the host computer before the application is deployed to the target. It locates all of the CalPoints in a target application and builds an internal table which is used during run-time. All of the CalPoints are checked for errors and any errors found are reported to the user within the dialog. The CalVIEW Master Configuration Dialog must be opened before deploying the target application if any CalPoints, Adapt Points, or Fault Points are added, removed, or modified since last opening the dialog. Failure to open the configuration dialog after making CalPoint changes will result in undefined behavior. The configuration dialog box opens automatically when the user drops the CalVIEW Master VI or when the user double clicks on the VI as shown in Figure 11.

The CalVIEW Master VI configuration dialog contains a target description and overall error indicator along the top of the dialog. There is a tab display in the center of the dialog containing summary information on the CalPoints that were found in the project. The bottom of the dialog contains an OK button that exits the configuration dialog box. In certain situations, the OK button changes to an Update Points button. The Update Points button causes CalVIEW Master to research the target and run the error checking again. This must be pressed before the OK button will reappear.



CalPoint Name	Type	Default	Units	Limits Enabled	Overwrite Cal	X Size	Y Size	ID	Description	Owning VI	Adaptable
CalPoint|Adapt|BOOLEnable	BOOLControl	F	|					41	This is a one shot to update	Target.vi	
CalPoint|Adapt|BOOLValue	BOOLControl	F	On|Off					42	This is the adapt point value	Target.vi	
CalPoint|Adapt|I32%	SGLControl	0.000000	%/100					43	This is the % to adapt the tal	Target.vi	
CalPoint|Adapt|I32Enable	BOOLControl	F	|					44	This is a one shot to update	Target.vi	
CalPoint|Adapt|I8%	SGLControl	0.000000	%/100					45	This is the % to adapt the tal	Target.vi	
CalPoint|Adapt|I8Enable	BOOLControl	F	|					46	This is a one shot to update	Target.vi	
CalPoint|Adapt|SGL%	SGLControl	0.000000	%/100					47	This is the % to adapt the tal	Target.vi	
CalPoint|Adapt|SGLEnable	BOOLControl	F	|					48	This is a one shot to update	Target.vi	
CalPoint|Adapt|SGLValue	SGLControl	0.000000	m					49	This is the adapt point value	Target.vi	
CalPoint|Adapt|TBL1D_SGL%	SGLControl	0.000000	%/100					50	This is the % to adapt the tal	Target.vi	
CalPoint|Adapt|TBL1D_SGLEnab	BOOLControl	F	|					51	This is a one shot to update	Target.vi	
CalPoint|Adapt|TBL1D_SGLX	SGLControl	0.000000	V					52	This is the adapt point value	Target.vi	
CalPoint|Adapt|TBL1D_SGLY	SGLControl	0.000000	kPa					53	This is the adapt point value	Target.vi	
CalPoint|Adapt|TBL2D_SGL%	SGLControl	0.000000	%/100					54	This is the % to adapt the tal	Target.vi	
CalPoint|Adapt|TBL2D_SGLEnab	BOOLControl	F	|					55	This is a one shot to update	Target.vi	
CalPoint|Adapt|TBL2D_SGLX	SGLControl	0.000000	V					56	This is the adapt point value	Target.vi	
CalPoint|Adapt|TBL2D_SGLY	SGLControl	0.000000	kPa					57	This is the adapt point value	Target.vi	

Figure 11: CalVIEW Master CalPoint Summary

The summary table contains a list of all the CalPoints found within the target along with most of their properties. The list is sorted in alphabetical order based on the CalPoints' full name. Double clicking on a CalPoint will open the CalPoint Configuration dialog box and allow changes to be made.

Importing a *.cdl file allows the user to apply default values from a calibration file to apply to the default values of the CalPoints. Exporting the table to a *.txt file, creates a tab delineated text file of the table. This table is used to simulate a target in the CalVIEW console in order to create a pairing file without a running target.

Below is a summary of each column and the information it contains.

CalPoint Name: Reports the names of the CalPoints.

Type: Reports the data type and direction of the CalPoint.

Default: Reports the default value that will be used by the CalPoint if no calibration file is found on the target, or if this particular CalPoint is not found in the calibration.

Units: Reports the units of the CalPoint entered in the configuration dialog.

Limits Enabled: Reports “T” if non-default limits are defined in the CalPoint configuration dialog.

Overwrite Cal: Reports “T” if the “Overwrite Cal” option is enabled within the CalPoint configuration dialog. This setting determines whether the default value is used during application startup, regardless of the value contained in the calibration file.

X Size: Reports the X size of a lookup table or array entered in the CalPoint configuration dialog.

Y Size: Reports the Y size of a lookup table or array entered in the CalPoint configuration dialog.

ID: Reports a unique CalPoint ID assigned to each CalPoint.

Description: Reports the description entered in the CalPoint configuration dialog.

Owning VI: Reports the VI name which contains the CalPoint. This is useful to debug any errors in the CalPoints.

Adaptable: Reports if an Adapt Point is set up to adapt the CalPoint.

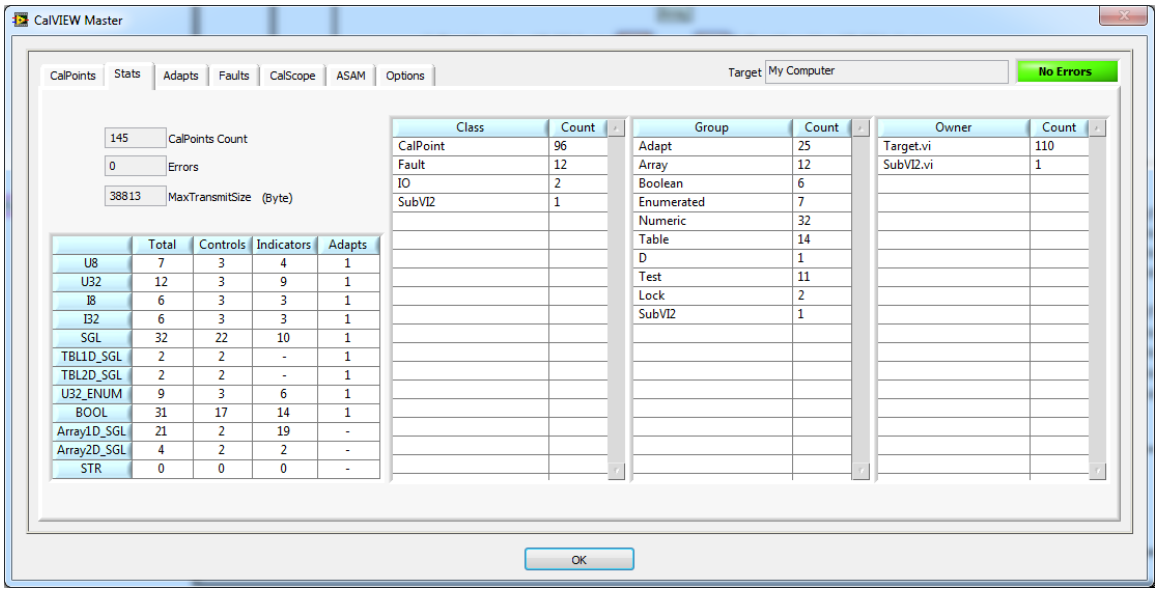


Figure 12: CalVIEW Master CalPoint Statistics

The Stats tab in the CalVIEW Master dialog contains a summary of CalPoint distributions. Distributions based on data type, group, class, and owning VI are reported. The maximum CalVIEW communication protocol packet size is displayed on this tab. The table build times are also shown with the total number of CalPoints and errors.

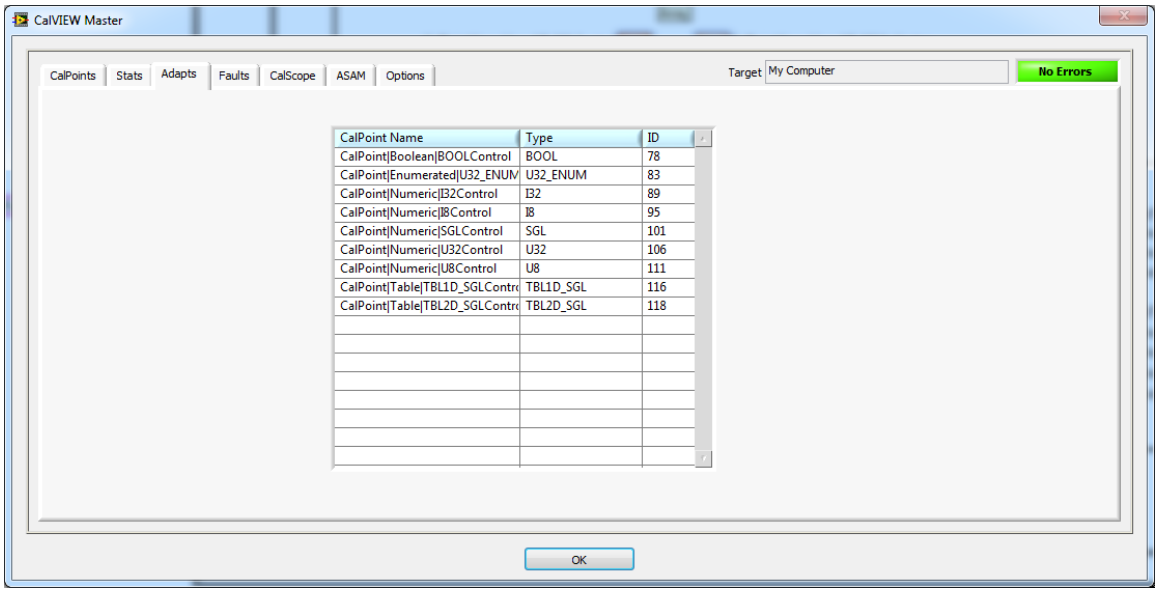


Figure 13: CalVIEW Master Adapt Point Summary

The Adapt tab displays a list of all the Adapt Points found in the project. The associated type and ID number are also displayed to allow easy identification of the associated CalPoint. Double clicking on a CalPoint will open the Adapt Point Configuration dialog box and allow changes to be made.

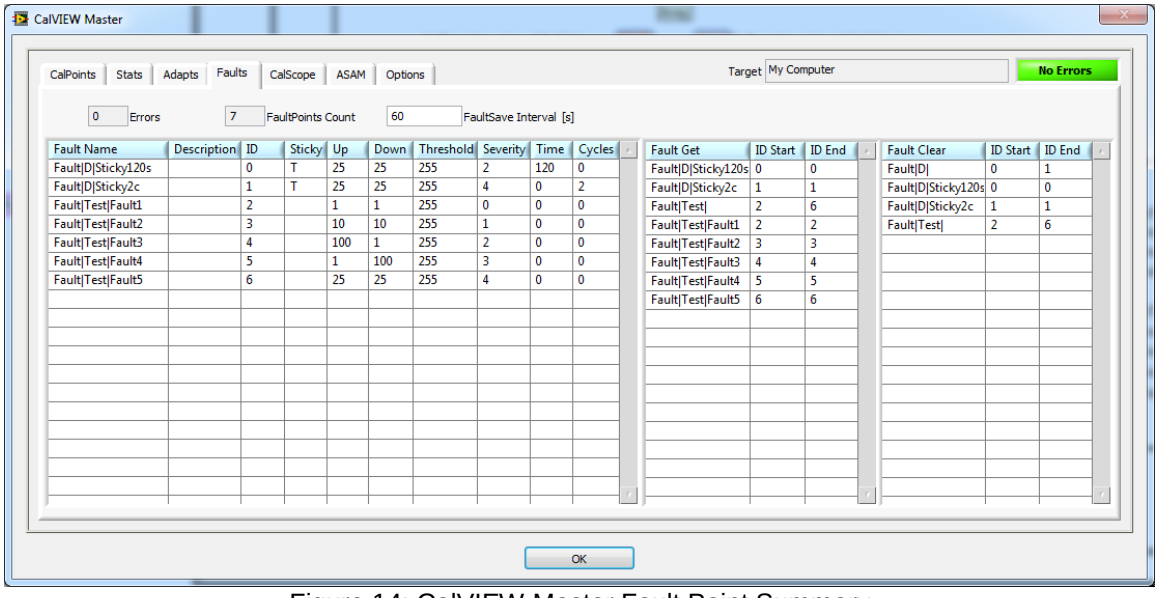


Figure 14: CalVIEW Master Fault Point Summary

The Fault Point tab in the CalVIEW Master dialog contains the summary of all fault points found on the target. Double clicking on a Fault Point will open the CalPoint Configuration dialog box and allow changes to be made.

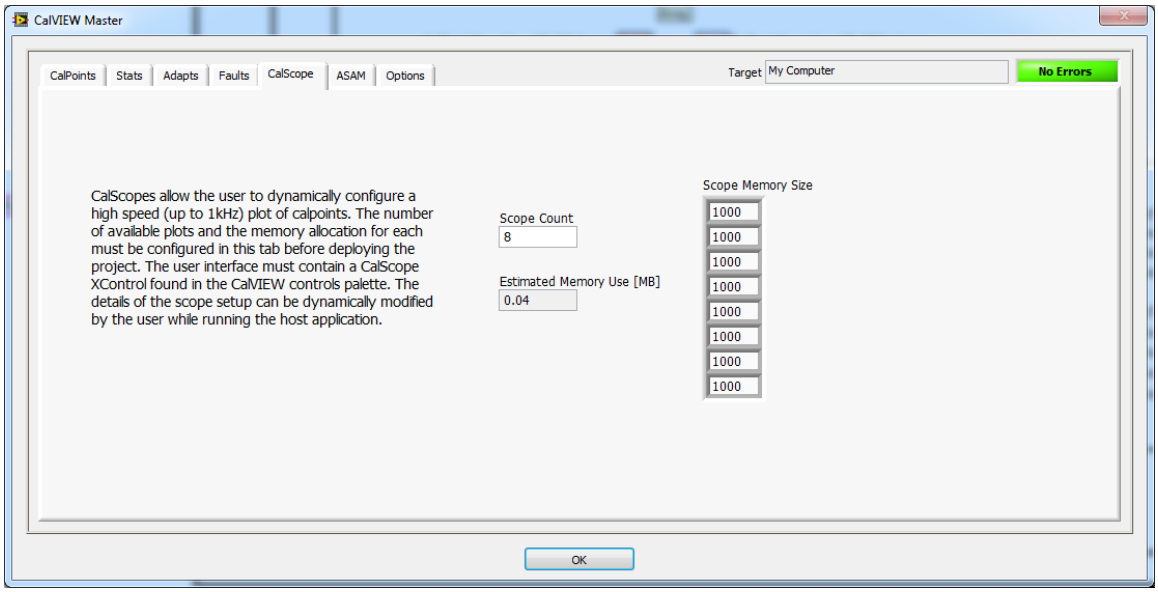

CalVIEW Master
CalPoints
Stats
Adapts
Faults
CalScope
ASAM
Options
Target
My Computer
No Errors
CalScopes allow the user to dynamically configure a high speed (up to 1kHz) plot of calpoints. The number of available plots and the memory allocation for each must be configured in this tab before deploying the project. The user interface must contain a CalScope XControl found in the CalVIEW controls palette. The details of the scope setup can be dynamically modified by the user while running the host application.
Scope Count
8
Estimated Memory Use [MB]
0.04
Scope Memory Size
1000
1000
1000
1000
1000
1000
1000
1000
OK

Figure 15: CalVIEW Master CalScope Setup

The CalScope tab contains the base setup for the CalScope controls. The user specifies the total number of scopes (up to 8) desired. Each scope also requires a dedicated memory buffer. All other CalScope configurations are handled during run-time by the user via the host user interface.

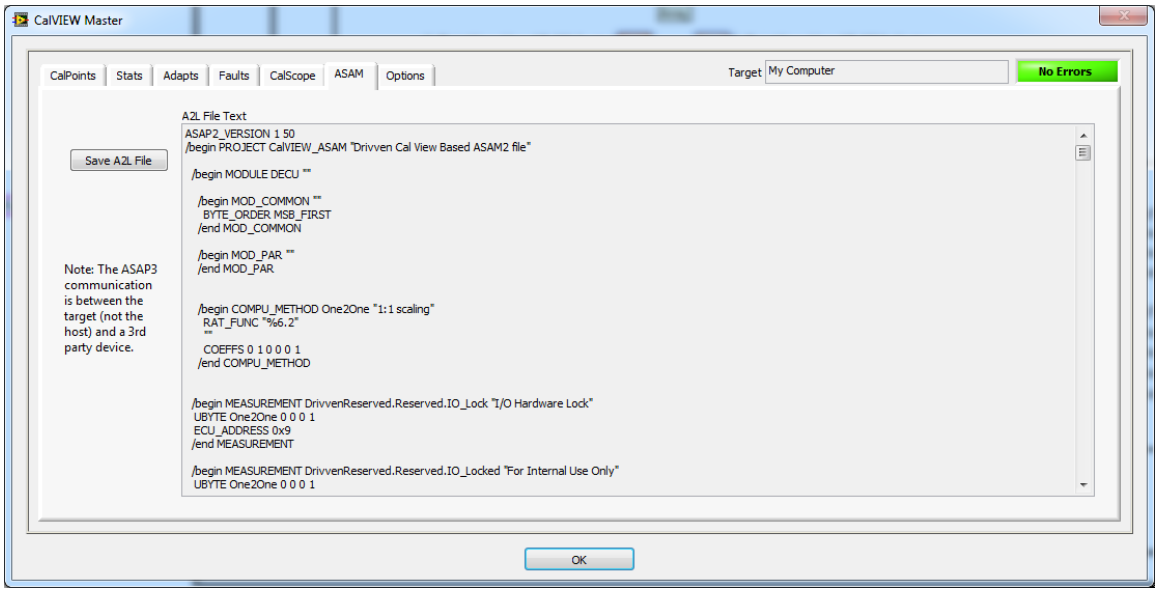
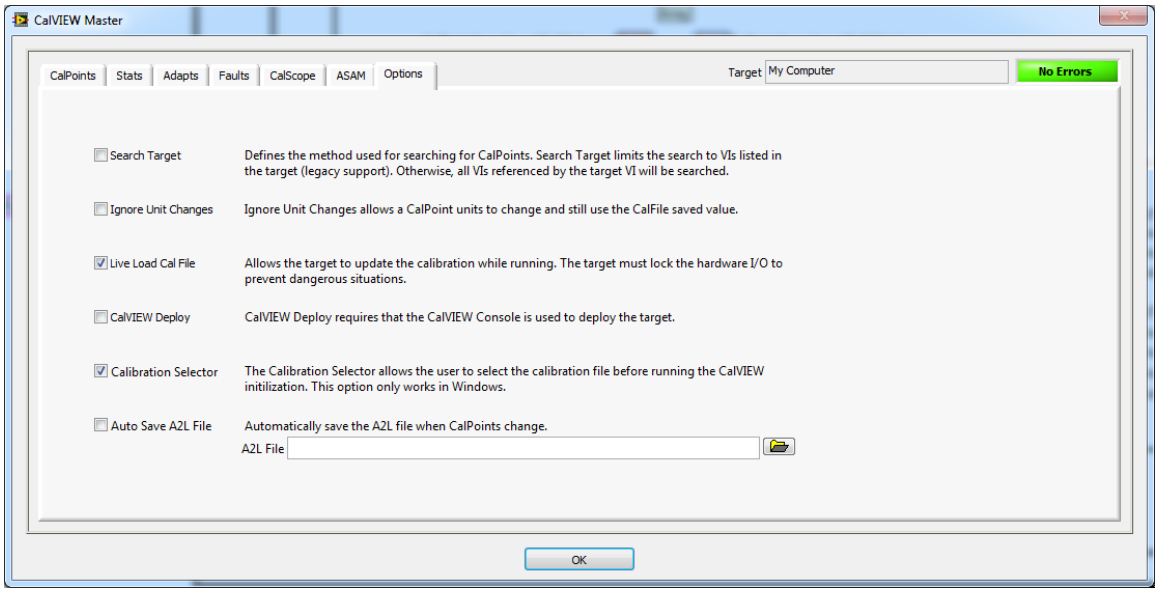


Figure 16: CalVIEW Master A2L File Setup

The CalVIEW Master dialog can export the CalPoint table to an A2L file. The A2L file can be imported to a test cell supervisory system which supports ASAP3 communication protocol. The CalVIEW run-time VI will listen for and respond to ASAP3 commands.

Note: The ASAP3 communication is between the target (not the host) and a 3rd party device.


CalVIEW Master
CalPoints
Stats
Adapts
Faults
CalScope
ASAM
Options
Target
My Computer
No Errors
Search Target
Defines the method used for searching for CalPoints. Search Target limits the search to VIs listed in the target (legacy support). Otherwise, all VIs referenced by the target VI will be searched.
Ignore Unit Changes
Ignore Unit Changes allows a CalPoint units to change and still use the CalFile saved value.
Live Load Cal File
Allows the target to update the calibration while running. The target must lock the hardware I/O to prevent dangerous situations.
CalVIEW Deploy
CalVIEW Deploy requires that the CalVIEW Console is used to deploy the target.
Calibration Selector
The Calibration Selector allows the user to select the calibration file before running the CalVIEW initilization. This option only works in Windows.
Auto Save A2L File
Automatically save the A2L file when CalPoints change.
A2L File
OK
Figure 17: CalVIEW Master Options

Search Target: The CalPoint search has 2 options to determine how CalVIEW finds CalPoints. The first option is to search targets. This option forces CalVIEW to search only the VIs added to the current target in the project. The alternative is to let CalVIEW search through all the referenced VIs even if they are not listed in the target. The other option is to include disabled structures. Commenting out sections of code will not change the list of available CalPoints.

Ignore Unit Changes: Selecting the Ignore Unit Changes button allows the units field from the calibration file to be used instead of those specified with the CalPoint configuration dialog in the case that they do not match. This option is always unchecked when the CalVIEW Master dialog opens. By default, changing the units field of a CalPoint causes the CalPoint to be treated as new in the calibration file, thereby losing the calibration value. However, the CalPoint value from the calibration file can be retained by selecting Ignore Unit Changes.

Live Load Cal File: Selecting Live Load Cal File enables the option to load a calibration file on a running target. The target must also confirm a hardware I/O lock before loading a calibration file.

CalVIEW Deploy: Selecting CalVIEW Deploy forces the user to deploy the VI from the CalVIEW consol. This guarantees that CalVIEW master runs before

deploying. If the target is deployed by pressing the run button, the target initialization will fail. Selecting this option will cause built applications to fail and should not be used with built applications.

Calibration Selector: Selecting Calibration Selector enables a calibration selection dialog box on target startup. This option only applies to windows targets.

AutoSave A2L: Selecting AutoSave A2L file causes the specified A2L file to save each time the CalVIEW Master searches for CalPoints.

The second function of the CalVIEW Master VI is to initialize the CalPoints and the Fault Points throughout the application and prepare for communications with the host. During application startup, the CalVIEW Master VI loads the calibration values from the specified “Calibration File” located in the file system of the target. The calibration file contains each CalPoint Name, Group, Class, Units, and Value. CalVIEW master also initializes the faults to their last known state. The state of the faults is constantly saved to a user specified file at a user specified rate. The file only contains the most recent stare of the faults.

During application startup, if the initialization finishes without any errors, the “InitOK” output is set to true. Only if the “InitOK” is true should any other part of the application be allowed to run. Allowing CalPoints to run with a false “InitOK” may cause the system behavior to be unpredictable. “InitOK” will be false if CalVIEW Master reports an error within the CalPoint table and the error is not resolved before running the application.

Specifying the Calibration File

The “Calibration File” input to the CalVIEW Master and Control CalVIEW VIs may be one of two formats. Those formats are referred to as Direct and Pointer formats.

The Direct format is to enter the calibration file name directly (with no file extension). As an example, if “DefaultCal” is used, then CalVIEW Master will always use the “DefaultCal.cdl” file located in the target’s “LabVIEW Data” directory. This method always uses the specified calibration file directly. To swap out calibration files using the direct method, then the application must be modified to specify the alternate file, or the alternate filename must be changed to match that of the currently specified file.

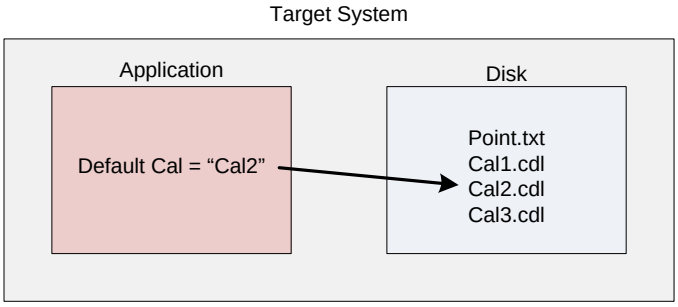


Figure 18: Direct Calibration File Format

The Pointer format is to enter a plain text filename, including the .txt extension. This plain text file should be located in the target’s “LabVIEW Data” directory and should point to the .cdl calibration file. Within the plain text calibration pointer file, the first line of text specifies the name of the .cdl calibration file to be used, leaving off the .cdl extension. This pointer method allows the user to swap calibration files by changing a simple text file instead of modify the application.

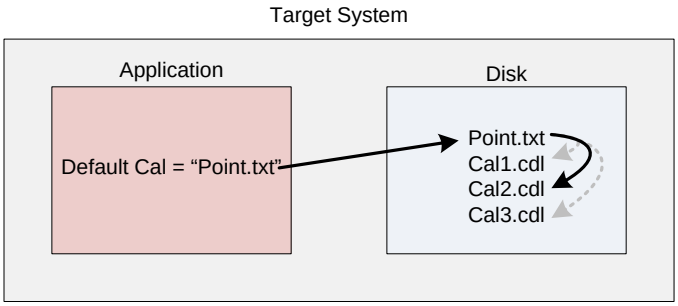


Figure 19: Pointer Calibration File Format

Calibration files are typically read during application startup. In the special case where Live Loading is enabled in the CalVIEW Master options, the calibration file may load while running. Live loading a calibration file doesn’t change the pointer file to use the loaded calibration. Due to the possible dangerous associated with loading a calibration file while running, several restrictions apply.

1. The application creator must enable live loading in CalVIEW Master.

2. The hardware I/O must be locked. The application creator must watch the “Request I/O Lock” and “Confirm the I/O Lock” when the hardware is in a safe state.

3. The user must understand and accept the risk of loading the calibration file on a running target.

CalPoints

Each CalPoint is a single point on the block diagram used to instrument the RT application. CalPoints are Express VIs and have a popup configuration dialog associated with them. Configuration dialog boxes will popup when a new CalPoint is dropped on the block diagram, or if the user double clicks on the CalPoint. Each CalPoint configuration dialog has a standard set of property fields associated with it. Some of the CalPoints have additional properties. Most CalPoints differ only in the fields for specifying the default value. On the block diagram, CalPoints are visibly different from standard indicator and controls by a light blue border around their icon and a light grey background color.

A list of all the CalPoints data types can be found in Table 2.

Table 2: CalPoint Data Type Summary

Type	Indicator	Control	Adaptable	Description
SGL	X	X	X	Single Precision Floating Point
I32	X	X	X	Signed 32-bit Integer
I8	X	X	X	Signed 8-bit Integer
U32	X	X	X	Unsigned 32-bit Integer
U8	X	X	X	Unsigned 8-bit Integer
BOOL	X	X	X	Boolean
1D Table		X	X	Single Precision 1D Lookup Table
2D Table		X	X	Single Precision 2D Lookup Table
1D Array	X	X		Single Precision 1D Array
2D Array	X	X		Single Precision 2D Array
STR	X	X		String
ENUM	X	X	X	Enumerated Unsigned 32bit Integer

Reserved CalPoints

CalVIEW automatically reserves a number of CalPoints at the beginning of the CalPoint list. These CalPoints contain properties of the target and host. They are also used for special CalVIEW functions, such as CalScopes. These CalPoints cannot be modified by the user. However, some of the reserved CalPoints may be connected to indicators on the host panel. The reserved CalPoints are in the “DrivvenReserved” class and “Reserved” group. User defined CalPoints are not allowed to be in the reserved class/group. The reserved CalPoints are saved in the calibration files. These may be helpful when reviewing CalFiles.

CalPoint Configuration Dialogs

Below is the common property fields associated with each CalPoint.

Name: CalPoint name.

Group: CalPoint group.

Class: CalPoint class.

Default Value: Value assumed at application startup if no calibration file is found, or if the named CalPoint is not found in the calibration file.

Overwrite Default Value: If True, the Default Value is used upon application startup, regardless of the calibration file value.

Units: String that will be displayed within the Caption of paired controls/indicators within the host VI.

Enable Limits: Enables limits to be imposed on CalPoint controls (Min, Max, Inc, Dec).

Min: Minimum CalPoint value (must be less than the maximum value).

Max: Maximum CalPoint value (must be greater than the minimum value).

MaxDiff: Maximum differential change in the CalPoint value (must be greater than zero).

MaxGrad: Maximum gradient allowed between Table CalPoints (must be greater than zero).

Description: Description of the CalPoint is used as the tip strip on the paired control/indicator within the host VI.

Each CalPoint must have a unique Class|Group|Name Combination. The “|” (pipe) character is used as a separator and should not be used in the Name, Class, Group or Units. Units with multiple parts use the “|” to separate the parts. The “.” character is also reserved. Spaces are permitted but not recommended.

Limits and slew rates for U32 and I32 CalPoints are represented internally as float32. Because of this not all 32 bit numbers are available, especially those that cannot be represented in less than 24 bits.

SGL, I32, I8, U32, U8 CalPoints

The CalPoint configuration dialog box in Figure 20 is representative of SGL, I32, I8, U32, and U8 data types (SGL shown). Limits are only applicable to controls.

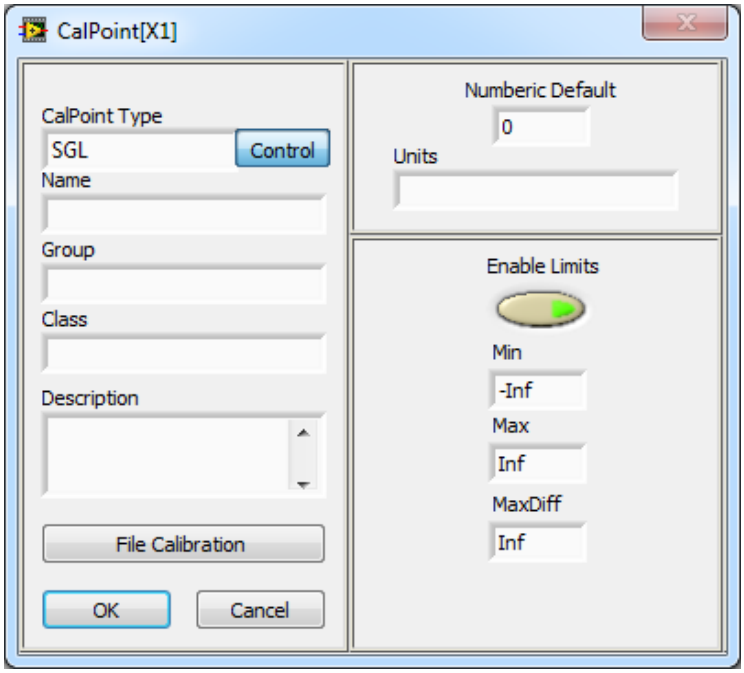


Figure 20: SGL Control Configuration Dialog Box

All other CalPoint types have a similar layout of properties, but have different interfaces for specifying the default values.

Array CalPoints

The Array CalPoint contains the property fields as shown in Figure 22. The array indicator and control CalPoint assumes a default value of zero and does not have a default value entry field. Arrays also have a maximum X/Y size that must be at least two. The X dimension always corresponds to the first dimension of a LabVIEW array. Values wired to an array CalPoint are not required to be the same size as the CalPoint. Larger arrays are cropped to the CalPoint size.

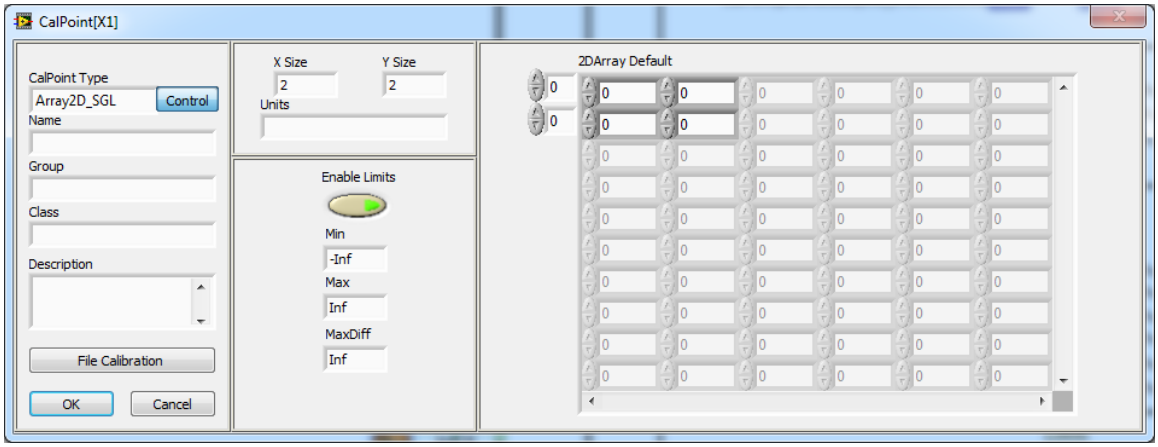


Figure 21: Array Configuration Dialog Box

String CalPoints

The String CalPoint contains the property fields as shown in Figure 22. A String CalPoint does not have units. Like the array CalPoints, the user must specify a maximum size. If the data is larger than the maximum size, the data is cropped at the maximum length.

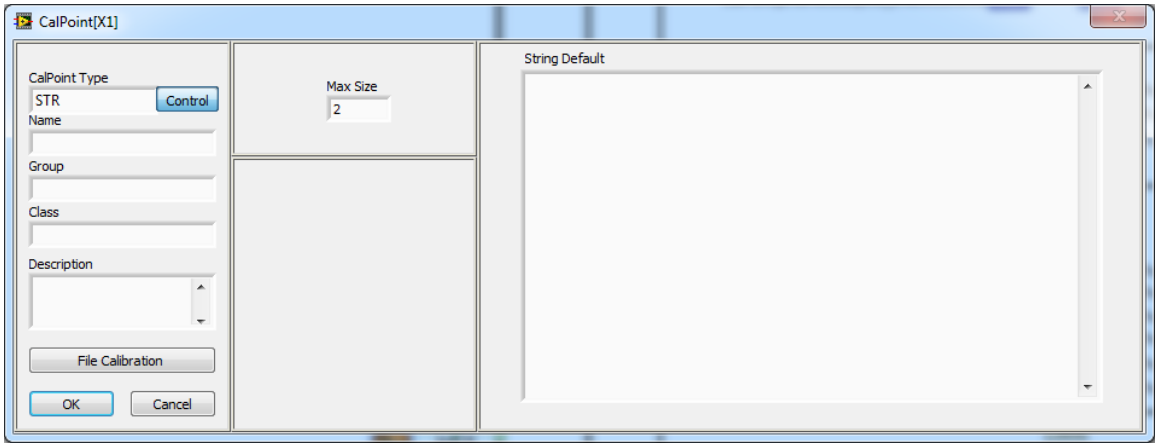


Figure 22: String Configuration Dialog Box

Boolean CalPoints

The Boolean CalPoint contains the property fields as shown in Figure 23. A Boolean CalPoint does not have units, but instead has True and False labels. The strings in the True and False fields are separated by a "|" in all summary displays. The strings entered for these fields show up as the Boolean text on the paired Boolean on the host VI front panel. Paired Boolean controls on the host front panel must be set to switch mechanical action and not latching. Boolean controls have a one shot function which provides a single true output after being set. The display control will also reset after the one shot fires.

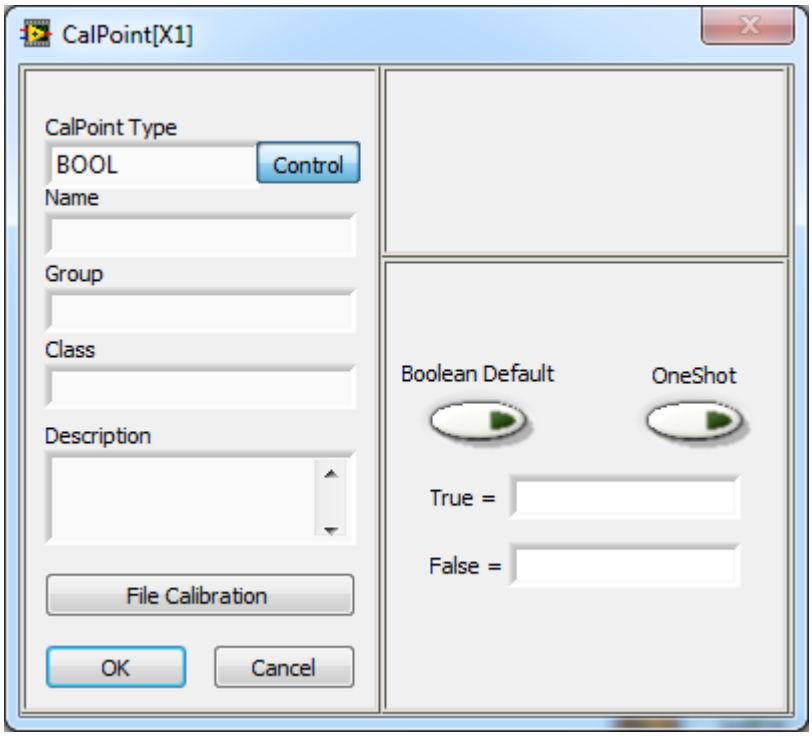


Figure 23: Boolean Configuration Dialog Box

Lookup Table CalPoints

The Lookup Table CalPoint contains the property fields as shown in Figure 23. The 1D and 2D lookup table units are split into 2 and 3 separate units fields, respectively. Each of the units fields correspond to a different part of the table and are separated by a "|" in all summary views. The default value area contains several parts. The X and Y indexing arrays for 1D and 2D table types must be monotonically increasing. They also must have a size of at least 2.

During runtime, the 1D lookup table outputs a value Y by interpolation according to the relative position of input X Index. During runtime, the 2D lookup table outputs a value Z by a three-way interpolation according to the relative position of inputs X Index and Y Index. If a value outside the table bounds is requested, the closest table value is used. The lookup table CalPoints can only be paired with a special CalVIEW table control on the Host VI, provided in the Windows Drivven→CalVIEW palette (front panel).

Figure 24: 1D Table Configuration Dialog Box

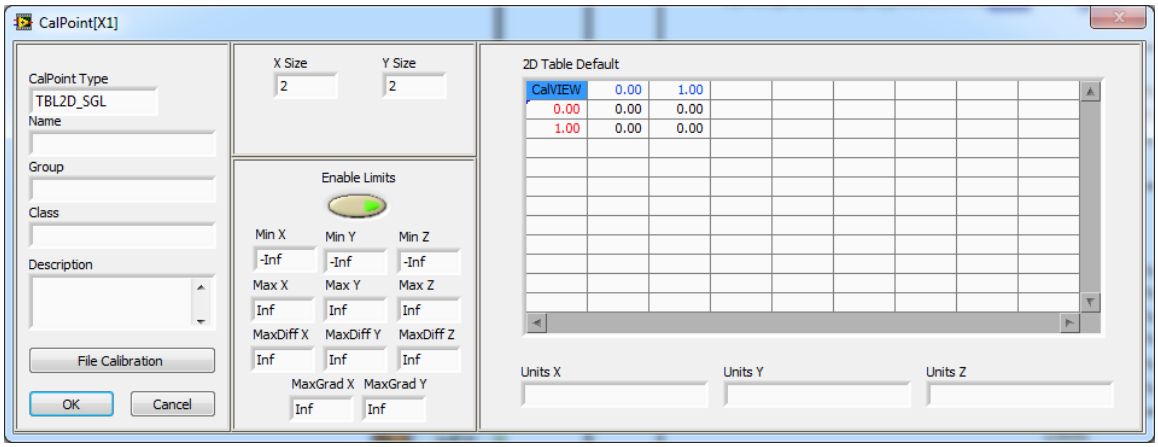


Figure 25: 2D Table Configuration Dialog Box

Enumerated CalPoints

The Enumerated CalPoint contains the property fields as shown in Figure 22. The enumerated CalPoint provides an array field for specifying the constant string associated with each possible integer value. The strings are separated by a “|” in all summary displays. The strings replace all of the values found on the paired host VI front panel enumerated controls/indicators. The real-time application requires that an integer, corresponding to the desired text value, be wired to the CalPoint. The paired enumerated item on the host VI must have the same number of enumerated items as the CalPoint. The string for each enumerated item on the host does not matter because the Text Labels from the target CalPoint will be copied over.

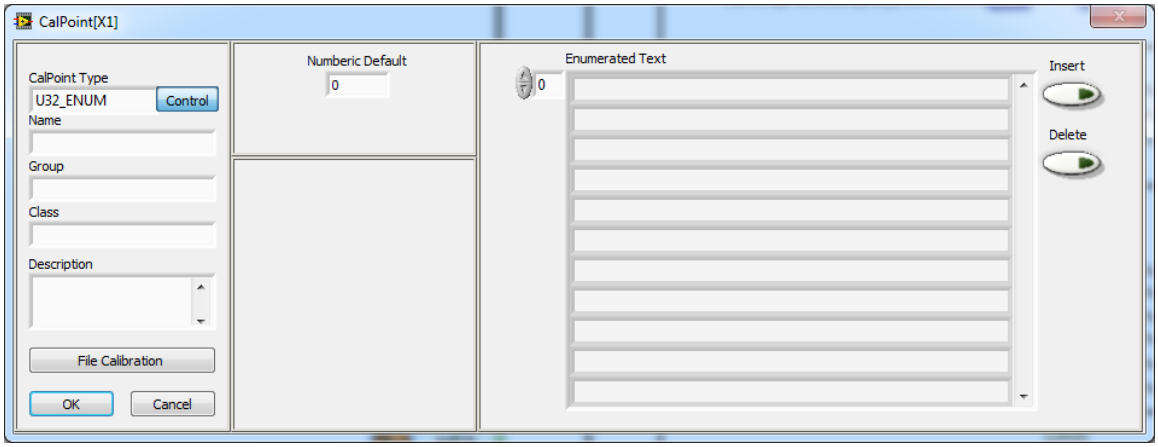


Figure 26: Enumerated Configuration Dialog Box

Adapt Points

With CalPoint controls by themselves, only the host user interface can modify their values. Adapt Points allow the target program to modify CalPoint controls. An Adapt Point can be placed in the block diagram and linked to a CalPoint control by assigning it the same Class|Group|Name. Each time the Adapt Point is written, the new value is loaded to the corresponding CalPoint control. The paired host control is also updated with the new value. The Adapt Points are found on the Adapt Points sub-palette on the main CalVIEW palette (Figure 6).

Adapt: When true, the corresponding CalPoint control will adapt. When false, nothing happens.

Value: New value to adapt the corresponding CalPoint control to. 1D and 2D CalPoint table controls have two and three inputs, respectively, for Value. They are “X” and “Y” for 1D tables and “X”, “Y” and “Z” for 2D tables.

%/100: Fractional adaptation allowed to the corresponding CalPoint. A value of 0 does not change the CalPoint. A value of 1 fully adapts the corresponding CalPoint to the value. A value of 0.5 adapts the corresponding CalPoint value ½ way between the old and new values.

Dropping an Adapt Point on the block diagram or double clicking on the block diagram icon opens the configuration dialog.

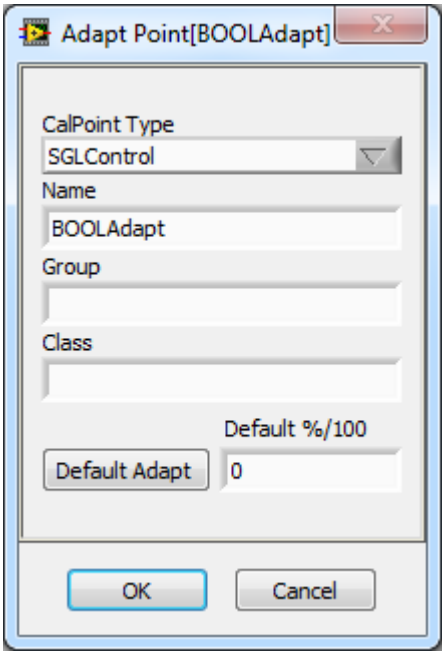


Figure 27: Adapt Point Configuration Dialog Box

CalPoint Type: Select the type of Adapt Point. This must match the desired corresponding CalPoint type.

Class, Group, Name: Adapt Point Class|Group|Name must match the desired corresponding CalPoint.

Default Adapt: Sets the default value for the “Adapt” input.

Default %/100: Sets the default value for the “%/100” input.

Fault Points

Fault points have 3 target operations. The first is to set the fault. The fault is based on an up/down counter which allows the programmer to specify a delay on setting and clearing the fault. The faults also have programmatic override option to disable the fault. A fault can also set sticky for a number of seconds or key cycles. A sticky fault will not clear until the up/down counter has been cleared for the specified time or key cycles. The second fault operation is to get a fault status. The get fault status works with a single fault, a group of faults, or a class of faults. The get fault status can also return only the unacknowledged faults. The last fault operation is to clear faults. The fault clear works with a single fault, a group of faults, or a class of faults. The fault clear can also acknowledge any unacknowledged faults.

Dropping an Adapt Point on the block diagram or double clicking on the block diagram icon opens the configuration dialog.

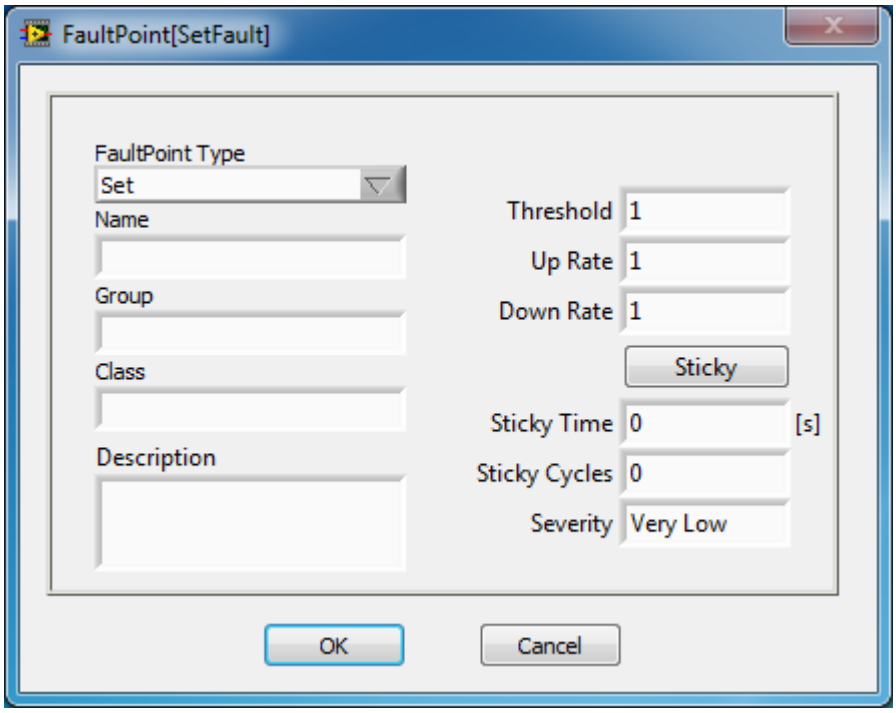


Figure 28: Fault Point Setup Dialog Box

FaultPoint Type: Select the type of Fault Point.

Class, Group, Name: Class|Group|Name are used in all types of Fault Point operations. For get and set Fault Points Class|Group|Name must match the desired corresponding Fault Point. Leaving the Name or the Group and Name blank will look at an entire group or class, respectively.

Description: Description of the Fault Point is displayed on the host VI fault XControl.

Threshold: The threshold is the value where the fault trips. The fault clears at zero.

Up Rate: The value at which the up/down counter increases each time the fault set is called and the input is set to true.

Down Rate: The value at which the up/down counter decreases each time the fault set is called and the input is set to false.

Sticky: Enable a sticky fault. The fault will not clear until the counter remains at zero for a specified time or number of key cycles.

Sticky Time: The amount of time required to clear a sticky fault.

Sticky Cycles: The number of key cycles required to clear a sticky fault.

Severity: The severity group defines what color and alarm sound are associated with the fault.

VxWorks Setup

All cRIO controllers developed by National Instruments after the original 9002/9004 controllers use the VxWorks operating system. Controllers with VxWorks operating systems will not automatically download the required shared library used by CalVIEW. Therefore, the user must ensure that the set up the target with the shared libraries.

The preferred way to set up CalVIEW on a VxWorks target is to use the Measurement and Automation eXplorer (MAX) (Figure 29). Right click on “Software” of the VxWorks target and select “Add/Remove Software”. Select the “Custom software installation” and add the current version of CalVIEW.

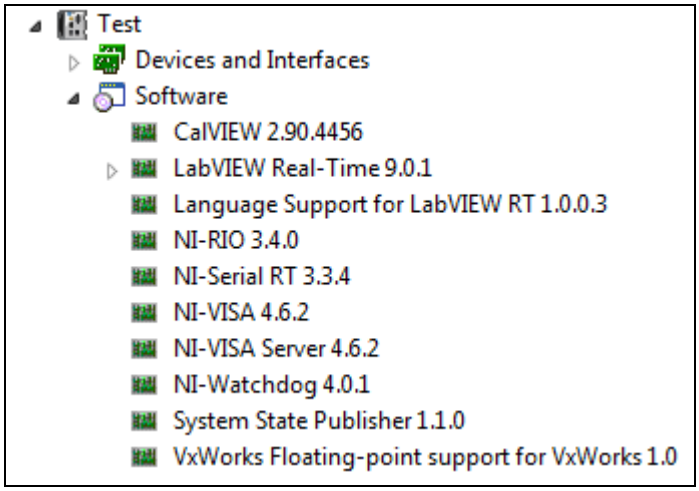


Figure 29: CalVIEW_C.out using MAX

Building Startup Executable Applications with CalVIEW

Applications which use CalVIEW can be built into a startup application similar to any other application. The user should remember to open the CalVIEW Master dialog before building the application executable so that all CalPoints are located. The user should also make sure that the CalVIEW_C.out file is on any VxWorks controllers before starting the application.

Backup and Restore of a Real-Time System

CalVIEW provides a resource for backing up and restoring real time targets. The tool uses the Real Time tools provided with the Real Time toolkit and is therefore compatible with other applications built using these tools. The

tool may be opened from Start→All Programs→Drivven→CalVIEW→RT Backup (Figure 30-Figure 32).

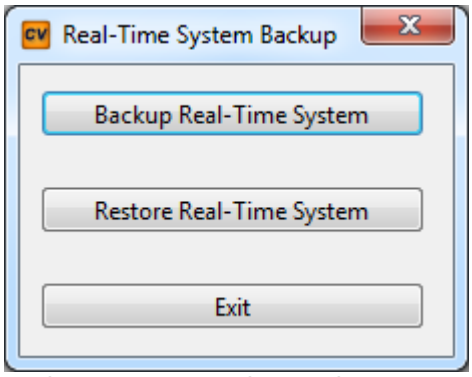


Figure 30: RT Backup and Restore

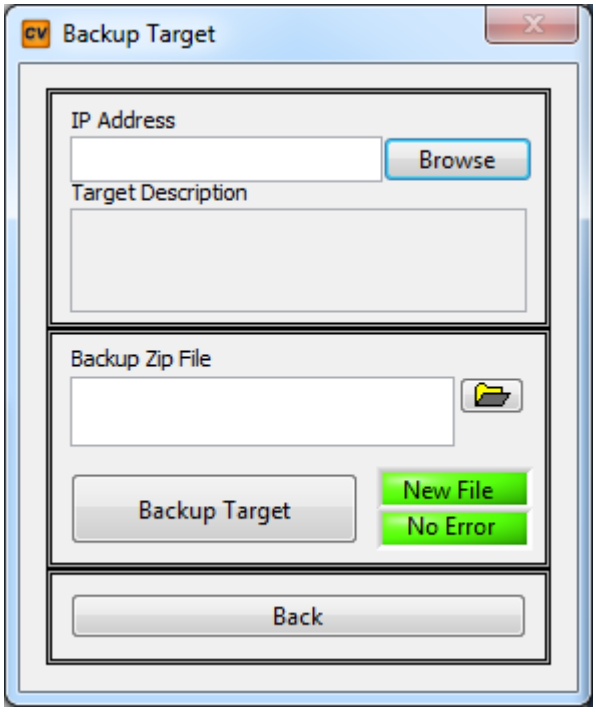


Figure 31: RT Backup Target

IP Address: Specifies the IP address of the target needing a backup.

Target Description: Displays information about the controller (ex. Type).

Backup Zip File: Specifies the Zip file to store the RT target image.

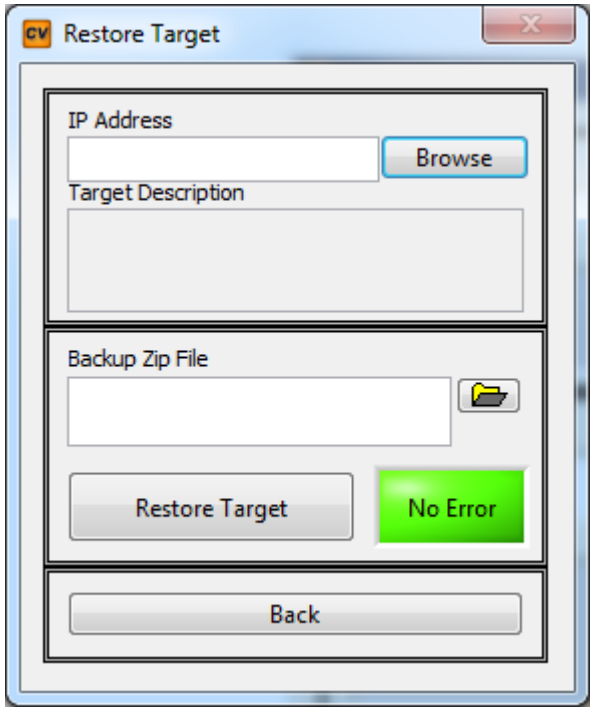


Figure 32: RT Restore Target

IP Address: Specifies the IP address of the target needing a restored.

Target Description: Displays information about the controller (ex. Type).

Backup Zip File: Specifies the Zip file that contains the RT target image.

Description: User comments on the RT target image.

CalVIEW Host

The host VI is a VI created by the project programmer and acts as the operator interface to the target. The main requirement for the host VI is that it must contain a while loop with a wait period. A wait period of 100 msec is recommended. The wait does not set the CalPoint update rate. It only effects the execution of the code in the host VI. It is not necessary to place anything else inside the loop. Then the programmer should create controls and indicators which match the data type of the CalPoints within the target application. It is not necessary for the names to be identical; however it does make the task of pairing target and host controls/indicators for communication easier. Control and indicator names on the host should only be used once. If the host VI contains two or more controls/indicators with the same name, CalVIEW console will only use the first one found. The controls and indicators in the block diagram of the host VI do not need to be wired to anything. They only need to exist. The CalVIEW console will enable you to make pairings between target CalPoints and these host controls/indicators. Then when you run the host VI from the CalVIEW console, the values will be communicated automatically between the host and the target. The block diagram shown in Figure 33 shows a simple host VI with columns of unwired controls and indicators. The code contains a simple loop with a wait command and a waveform chart for a few variables. The CalPoints will update at the update rate set in the CalVIEW console but the chart will update at the rate of the loop.

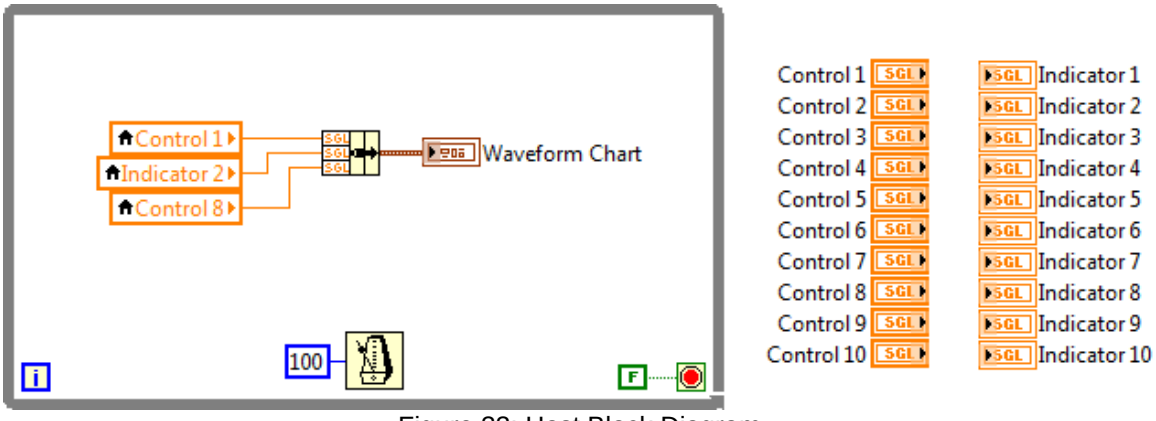


Figure 33: Host Block Diagram

Indicators in the host VI are only updated with CalPoint indicator values from the target when they are visible on the host VI front panel. Therefore if you have a tabbed display on the host front panel with indicators within the tabs, then only the visible tab will have its indicators updated. This helps to preserve communications bandwidth and lowers the target processor demand. If you have a need to write additional LabVIEW code within the host VI which utilizes CalPoint data (for example logging values to file), then you need to determine if

that value will be visible at all times. If it is within a tab, then it will be necessary to create an additional indicator which can be placed off-screen (but technically always visible). You can make an additional pairing between the CalPoint of interest and this off-screen indicator. Then you can create a local variable of this always-visible indicator and use the local variable for any additional LabVIEW coding.

CalPoints may be programmatically modified on the host using the Value(signaling) property node. (Figure 34) Using the Value property node or local variables will change the value on the user interface but it will not cause the value to be written to the target.

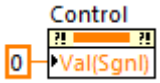


Figure 34: Value (signaling) Property Node

When changes are made to the host VI, the CalVIEW console needs to be refreshed in order to be made aware of those changes.

When the “Start Host VI” button is pressed in the CalVIEW console, a run-time version of the host VI is opened. The block diagram of this run-time version is not accessible. Any programmatic changes to the host VI must be performed by opening an editable version manually from disk or from the project explorer.

Lookup Table Controls

CalVIEW uses custom XControls to display Lookup Tables on the host VI. The XControl allows advanced viewing and editing options. The control displays the current input and output values along with the lookup table with the current value highlighted. In addition, the user may directly edit the table values or use a set of built-in functions. A plot of the data is also available. The table controls can be found in the Drivven→CalVIEW palette on the front panel. To view a larger table, simply drag the corners of the control and more cells will be added as room becomes available. Horizontal and vertical scroll bars may also be enabled for large tables to be displayed in a small space.

1D Tables

1D tables consist of a set of matching X and Y arrays. The X values correspond to the input and the Y values correspond to the output values. Figure 35 shows the basic 1D table. Several options are available by right clicking on the table.

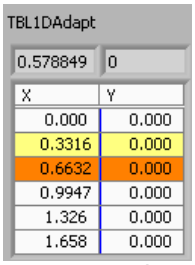


Figure 35: 1D Lookup Table

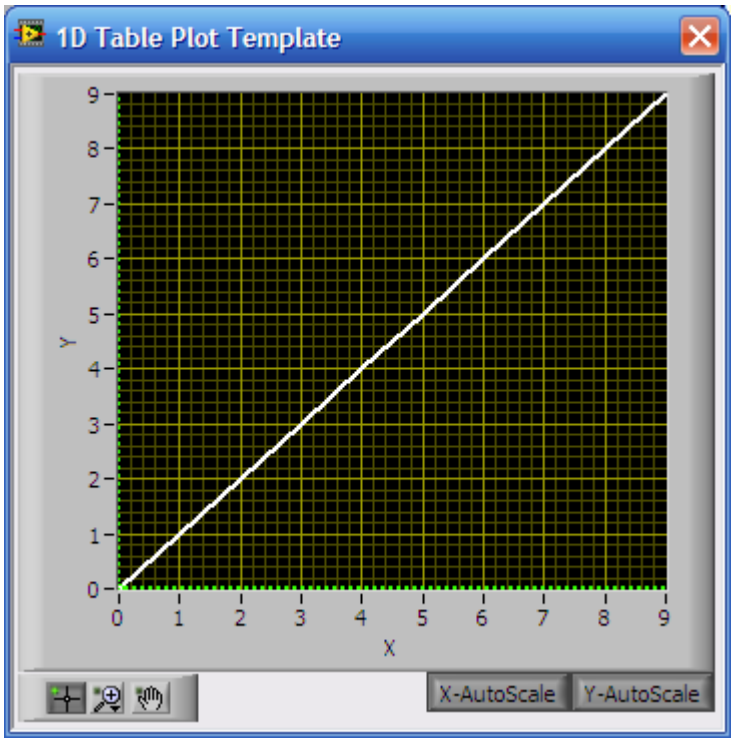


Figure 36: 1D Table Plot Window

The following options are available by right clicking on the 1D table control.

Copy Data: Copy the table data to the clipboard in a spreadsheet format.

Insert Data: Opens the insert data dialog.

Undo: Revert the table data to the previous state.

Modify Format String: Changes the format of numbers displayed in the table.

Enter a valid format string. Please see www.ni.com for details on “Format Specifier Syntax Elements”

%	Starts format string
+	Always show sign (optional)
^	Exponent is always a multiple of 3 (optional)
#	Removes trailing zeros (optional)
0	Pad to left with zeros (optional)
x	Field width (optional)
_y	y Digits of precision (optional)
or	
.y	y Significant digits (optional)
e	Floating point number with fractional format
or	
f	Floating point number with scientific notation
or	
g	Uses e or f depending on the case
or	
p	Floating point number in SI notation

Example
12.345

%.2f	=	12.35
%_3f	=	12.3
%.5f	=	12.34500
%#.5f	=	12.345
%+f	=	+12.345
%e	=	1.234500E+1

Open Plot Window: Opens a popup window with a plot of the table and the current value.

Show In and Out Values: Displays the current X input and Y output values from the target at the top to the table.

Show Highlight: Highlights the table value used to calculate the output value. All of the values used are highlighted in yellow except the closest value. It is highlighted in orange.

Show Limits: Shows the value of the limits as well as the current value found in the table.

Scale Selection: Scale the current selection by a user specified value. (ex. [2,4,6,11] x 2 = [4,8,12,22])

Scale Difference of Selection: Scale the difference between the first selected value and all the rest of the selected points by a user specified value. The first value will not change. This function is equivalent to subtracting the first selected value from the array, scaling the array and then adding the original value to the array. This may be useful if the slope of the table needs to change but the starting point is acceptable. (ex. [2,4,6,11] @ 2x → [2,6,10,20])

Straighten Selection: Sets the Y values so that they are defined by a straight line (Y = mX + b) between the first and last value. This function depends on the value of X for each point. The output of this function will be a straight line on the table plot.

Set Selection Value: Sets all the selected values to a user specified value. (ex. [2,4,8,11] @ 2 = [2,2,2,2])

Space Evenly: Sets the values of all the selected points so that the difference between any two consecutive points is equal. The first and last selected values are the only values that affect the final values and do not change. This function returns the same results as the straighten selection if the index values are evenly spaced. (ex. [2,4,6,11] → [2,5,8,11], (11-2)/(4-1)=3, a difference of 3 between each point)

Vertical Scrollbar: Adds a vertical scrollbar to the table.

2D Tables

2D tables consist of a set of interacting X, Y and Z arrays. The X and Y values correspond to the inputs and the Z values correspond to the output values. Figure 37 shows the basic 2D table. The X values are shown in red, the Y values are shown in blue and the Z values are shown in black. Several options are available by right clicking on the table.

TBL2DAdapt

5 12 8.5

CalVIEW	0.000	20.00	40.00	60.00	80.00	100.0
0.000	0.000	10.00	20.00	30.00	40.00	50.00
20.00	10.00	20.00	30.00	40.00	50.00	60.00
40.00	20.00	30.00	40.00	50.00	60.00	70.00
60.00	30.00	40.00	50.00	60.00	70.00	80.00
80.00	40.00	50.00	60.00	70.00	80.00	90.00
100.0	50.00	60.00	70.00	80.00	90.00	100.0

Figure 37: 2D Lookup Table

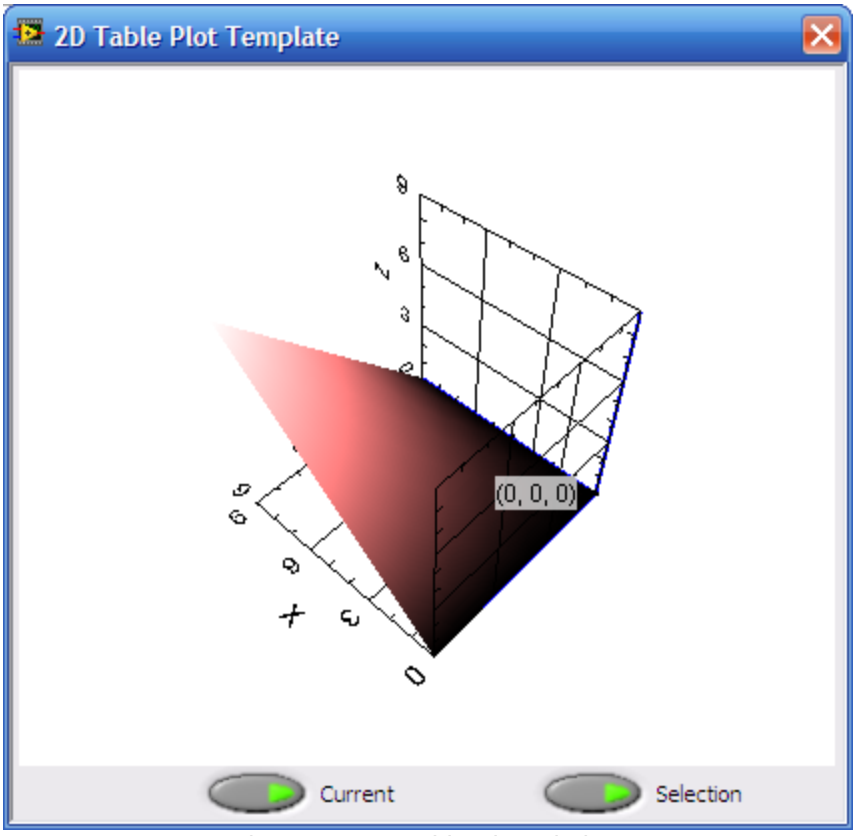


Figure 38: 2D Table Plot Window

Copy Data: Copy the table data to the clipboard in a spreadsheet format.

Insert Data: Opens the insert data dialog.

Undo: Revert the table data to the previous state.

Modify Format String: Changes the format of numbers displayed in the table.

Open Plot Window: Opens a popup window with a plot of the table and the current value.

Show In and Out Values: Displays the current X and Y inputs values and the Z output value from the target at the top to the table.

Show Highlight: Highlights the table value used to calculate the output value. All of the values used are highlighted in yellow except the closest value. It is highlighted in orange.

Show Limits: Shows the value of the limits as well as the current value found in the table.

Scale Selection: Scale the current selection by a user specified value. (ex. [2,4,6,11] x 2 = [4,8,12,22])

Scale Difference of Selection: Scale the difference between the top left selected point and all the rest of the selected points by a user specified value. The top left value will not change. This function is equivalent to subtracting the top left selected value from the array, scaling the array, and then adding the original top left value to the array. This may be useful if slope of the table needs to change but the starting point is acceptable. (ex. [2,4,6,11] @ 2x → [2,6,10,20])

Flatten Selection: Sets the Y values so that they can be defined by a flat surface between the four selected corners. This function depends on the value of X and Y for each point. The output of this function will be a flat surface on the table plot.

Set Selection Value: Sets all the selected values to a user specified value. (ex. [2,4,6,11] @ 2x → [2,2,2,2])

Space Evenly: Sets the values of all the selected points so that the difference between any two consecutive points is equal. The corner values are the only values that affect the final values and do not change. This function returns the same results as the Flatten Selection if the index arrays are evenly spaced. (ex. [2,4,6,11] → [2,5,8,11], (11-2)/(4-1)=3, a difference of 3 between each point)

Vertical Scrollbar: Adds a vertical scrollbar to the table.

Horizontal Scrollbar: Adds a horizontal scrollbar to the table.

CalScope

CalScopes allow the user to dynamically setup and run CalPoint plots. The number of plots and their associated memory use must be configured in CalVIEW Master from the RT target block diagram prior to running the application. A CalScope XControl must be placed on the host VI front panel. The CalScope XControl (Figure 39) is found in the Drivven→CalVIEW palette menu. Only one copy of the CalScope XControl may be use in a host VI.

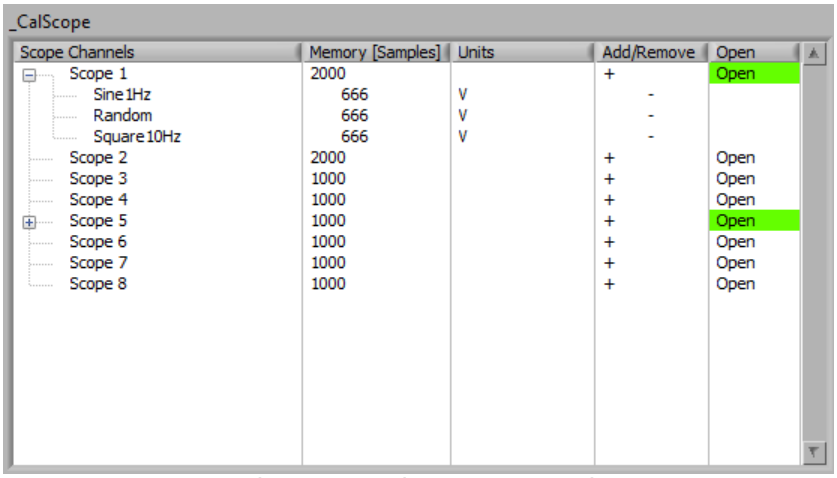


Figure 39: CalScope XControl

Clicking in the “Scope Channels” column allows the user to change the name of the scope or change the CalPoint. The “Memory” column shows the number of samples allocated for each scope and for each channel in a scope. The memory is set using the CalVIEW Master configuration dialog box. The “Units” column displays the units of the CalPoint as entered in the CalPoint configuration dialog in the RT block diagram. The “Add/Remove” column allows the user to add a CalPoint to the scope by pressing the + button. Clicking the - button will remove the CalPoint. The “Open” column allows the user to open the scope window. Scopes that are configured correctly and enabled for opening will have a green background.

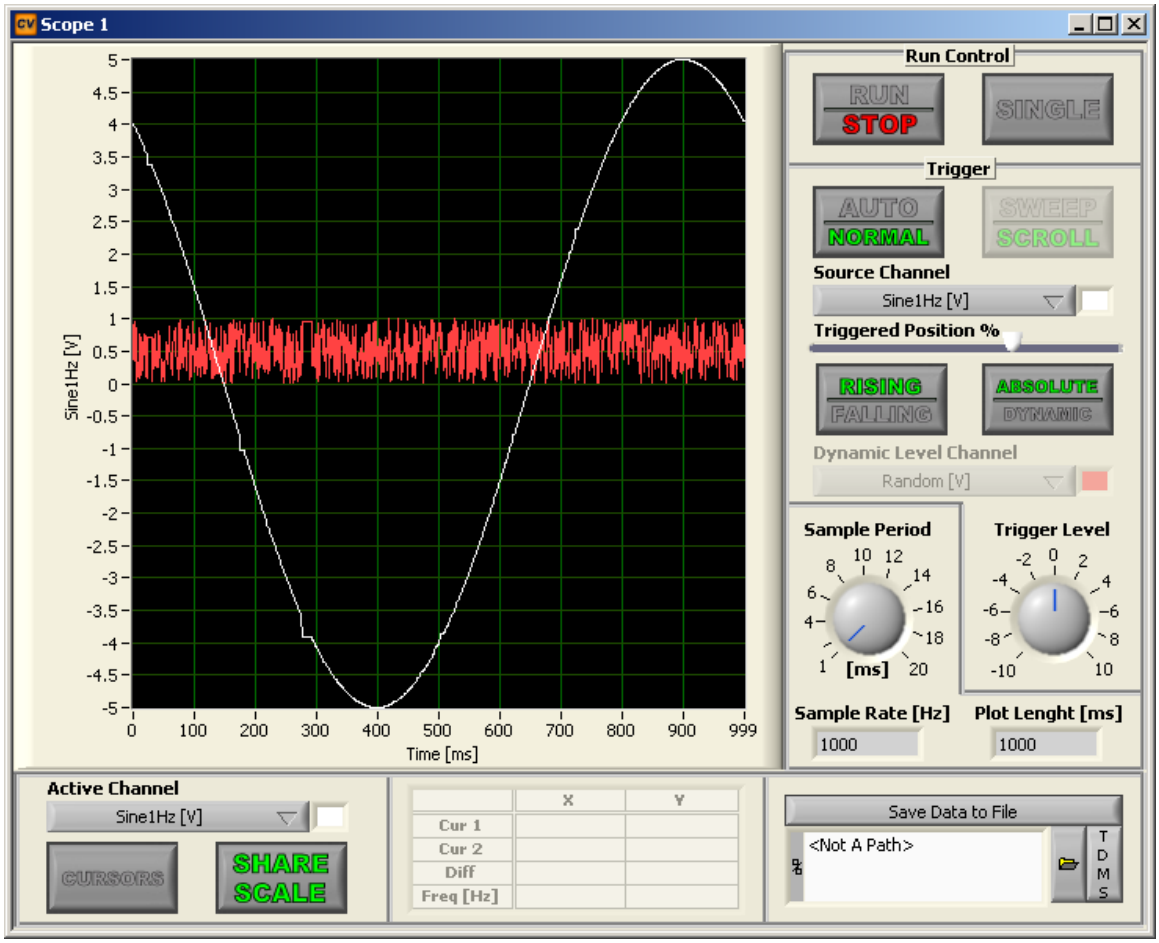


Figure 40: CalScope

The CalScope plot window contains the rest of the scope setup and control.

Active Channel: Selecting a CalPoint from the Active Channel drop-down list will display its scale on the left side of the plot. The cursors are linked to the active channel. The color box on the side of the active channel drop-down list sets the line color of the active channel.

Cursors: Cursors are turned on and off using this button. When active, two cursors are displayed on the plot. The user can select and move the cursors in the scope window. The cursor locations are displayed to the right, along with the differences and the frequency of the cursors.

Share Scale: This control links the scales of all channels. When shared, all the scales are the same regardless of the units. When not shared, only the active channel scale is displayed. Each scale is configured separately.

Save Data To File: Saves the current plot capture to the specified file. Files may be stored in a TDMS or CSV format according to the TDMS/CVS button text shown.

Run/Stop: When in Run mode, CalPoints are sampled according to the current trigger configuration. When in Stop mode, all sampling is stopped.

Single: Captures a single waveform according to the current trigger configuration and then stops.

Auto/Normal: Auto mode continually captures a waveform regardless of the selected trigger source. In auto mode, all other trigger options are disabled. In Normal mode, data is captured according to the current trigger configuration. The location of the trigger is shown by a vertical cursor.

Sweep/Scroll: Controls the way the auto mode displays data in the scope window. The display will scroll data or sweep the new data.

Source Channel: Sets the CalPoint used to trigger a new data capture.

Triggered Position %: Sets the fraction of the buffer used for pre-trigger data.

Rising/Falling: Sets the direction of which the trigger source must pass through the trigger level to trigger a new data capture.

Absolute/Dynamic: In Absolute mode, Trigger Level is used as the absolute trigger threshold. In Dynamic mode, the CalPoint selected in the Dynamic Level Channel drop-down list is used as the dynamic trigger threshold.

Dynamic Level Channel: Sets the CalPoint used as the dynamic trigger threshold.

Trigger Level: Sets the absolute trigger threshold for the Source Channel.

Sample Period: Determines the sample period in milliseconds of the data collection. Sample Period must be in increments of 1ms. The range may be modified by double-clicking and editing the dial values.

Sample Rate: Displays the calculated sample rate in hertz of the data collection according to the selected Sample Period.

Plot Length: Displays the calculated length of the data collection and plot window in milliseconds.

CalTrend

CalTrends provide plotting and configuration similar to CalScopes. The primary difference is that CalTrends do not need to be configured in CalVIEW Master and they operate at a maximum sample rate of 1Hz. Only one copy of the XControl may be used on a host VI front panel. Only CalPoints paired to host items may be used in a trend.

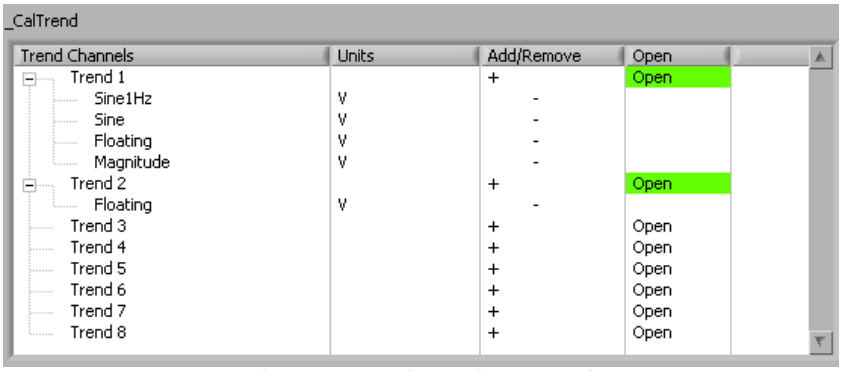


Figure 41: CalTrend XControl

The CalTrend XControl allows the user to configure up to 8 trend charts. Clicking in the “Trend Channels” column allows the user to change the name of the trend or change the CalPoints. The “Units” column displays the units of the CalPoint and may be affected by the unit conversion utility within CalVIEW. The “Add/Remove” column allows the user to add a CalPoint to the trend by pressing the + button. Clicking the - button will remove the CalPoint. The “Open” column allows the user to open the chart window. Trends that are configured correctly and enabled for opening will have a green background.

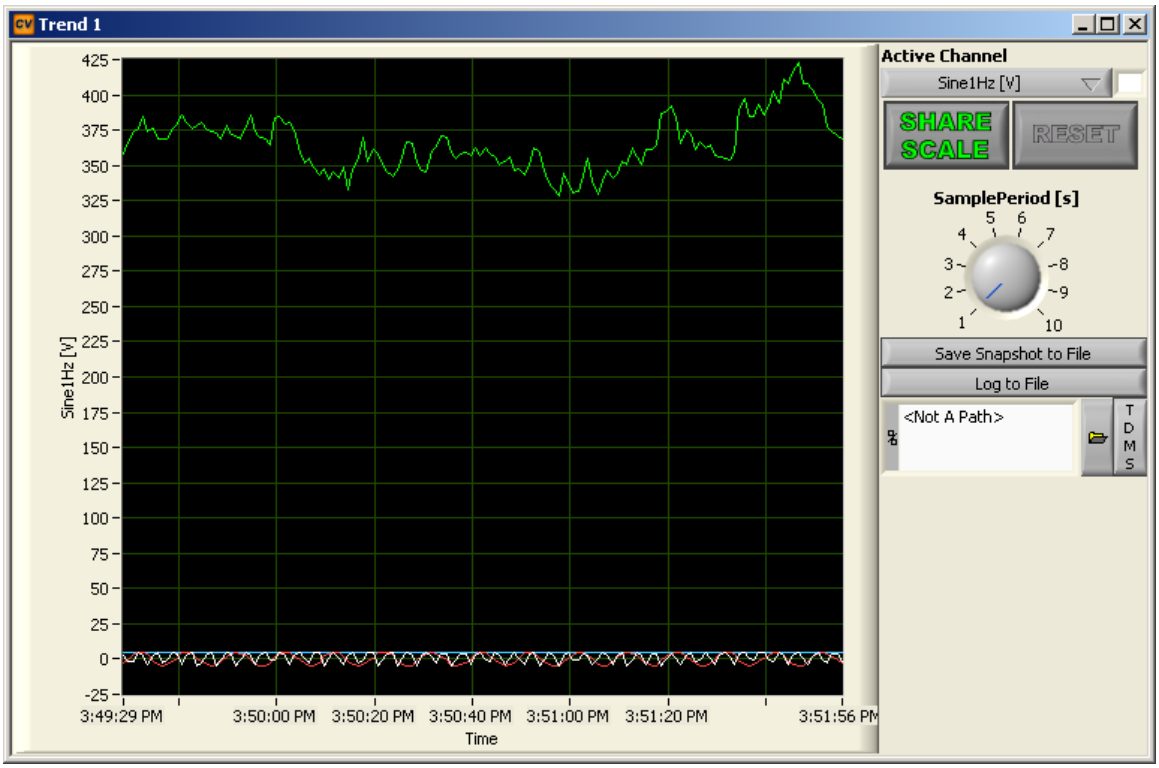


Figure 42: Trend Plot

The trend popup window provides all the remaining configurations for the trends.

Active Channel: Selecting a CalPoint from the Active Channel drop-down list will display its scale on the left side of the trend. The cursors are linked to the active channel. The color box on the side of the active channel drop-down list sets the line color of the active channel.

Share Scale: This control links the scales of all channels. When shared, all the scales are the same regardless of the units. When not shared, only the active channel scale is displayed. Each scale is configured separately.

Reset: Manually clears all the accumulated data and starts the trend over.

Sample Period: Determines the sample period in seconds of the data collection. Sample Period must be in increments of 1ms. The range may be modified by double-clicking and editing the dial values.

Sample Period: Determine the sample rate of the data collection. These must be in increments of 1s.

Save Snapshot To File: Saves the current plot to the specified file. Files may be stored in a TDMS or CSV format according to the TDMS/CVS button text shown.

Log to File: Continually logs new data to the specified file. Files may be stored in a TDMS or CSV format according to the TDMS/CVS button text shown.

Faults

Faults are displayed and control using the Faults XControl. The user simply needs to add a single Faults XControl to the host VI and the console will automatically connect it. All the faults will display on the XControl in alphabetical order. The user may filter out all the inactive faults and/or all faults below a certain severity level. Double clicking on the fault name will first acknowledge and then clear the fault. Double clicking on the trip count of a fault will clear that fault's trip count. Double clicking on the disable of a fault will cause the fault to disable until the disable is cleared by the user.

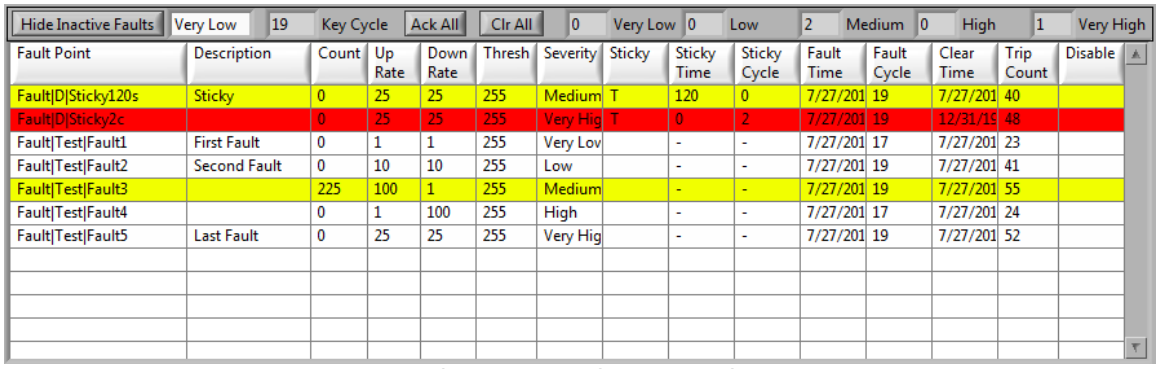


Figure 43: Faults XControl

Adding faults to a host VI also adds an indicator to the console to show the state of the faults. Each fault level is represented by a different color. If alarms are active, the color will display. If an alarm has not been acknowledged, the indicator will flash and the alarm will sound if enabled.

Figure 44: Fault Indicator

Sub-Hosts

Sub-hosts are defined using the sub-hosts XControl. They follow the same rules as a top level host VI. Right clicking on the XControl bring up configuration options for setting the host VI, pairing file, and display icon. The user can also specify a display name and one or more display groups. While running, clicking the XControl opens the sub-host.

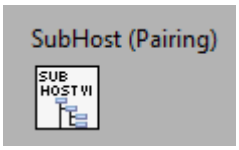


Figure 45: Sub-Host XControl

A Sub-Host List XControl is provided to display a list of all sub-hosts in the host VI and their state. No user setup is required after placing it on the host VI front panel. The sub-host List may be assigned to a group so that it only sub-hosts in the specified group. Double clicking on the sub-host name will open or close the sub-host VI. Open sub-hosts names are bold.

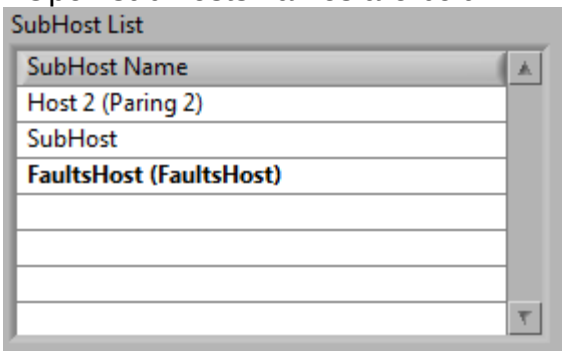


Figure 46: Sub-Host List XControl

Other Host VI Tools

The CalVIEW palette contains several XControls to help make powerful and easy to setup user interfaces. These XControl were specifically designed to work as host VI items connected to CalPoints.

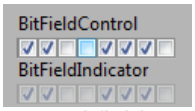


Figure 47: Bitfield XControl

The Bitfield XControl allows the host to display and enter data to a U32 CalPoint one bit at a time. It is valid as a control or indicator. The number of bits shown can be modified by changing the size of the control. The control may also be displayed vertically.

Figure 48: Bitfield XControl

The Path XControl allows the host to enter a path but connect it to a string CalPoint. It is only valid as a control. When providing an IP address of the target, it will determine if the target is the local or remote computer and use the windows file browser or FTP when appropriate. This tool may not work in the

LabVIEW development environment if the LabVIEW Internet Toolkit is not installed.

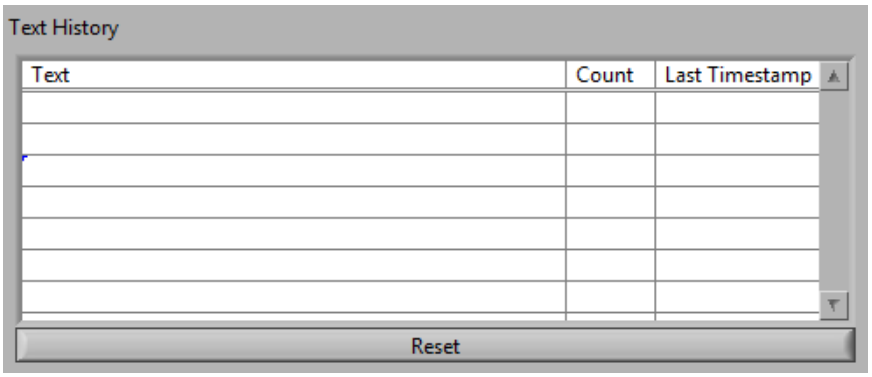


Figure 49: Text History XControl

The Text History XControl can pair with string CalPoint. It captures and holds text that may flash on the screen too fast for the user to read.

CalVIEW Console

The CalVIEW console application for Windows can be found at Start→Programs→Drivven→CalVIEW. The CalVIEW console opens the display shown in Figure 50 (full mode) and Figure 51 (mini mode). The CalVIEW program is the communication link between the host VI and the real-time target. Multiple instances of the CalVIEW console may be opened on a single computer to allow communication with multiple targets. The instances may be opened using the start menu or using the CalVIEW Settings File (*.cvsf).

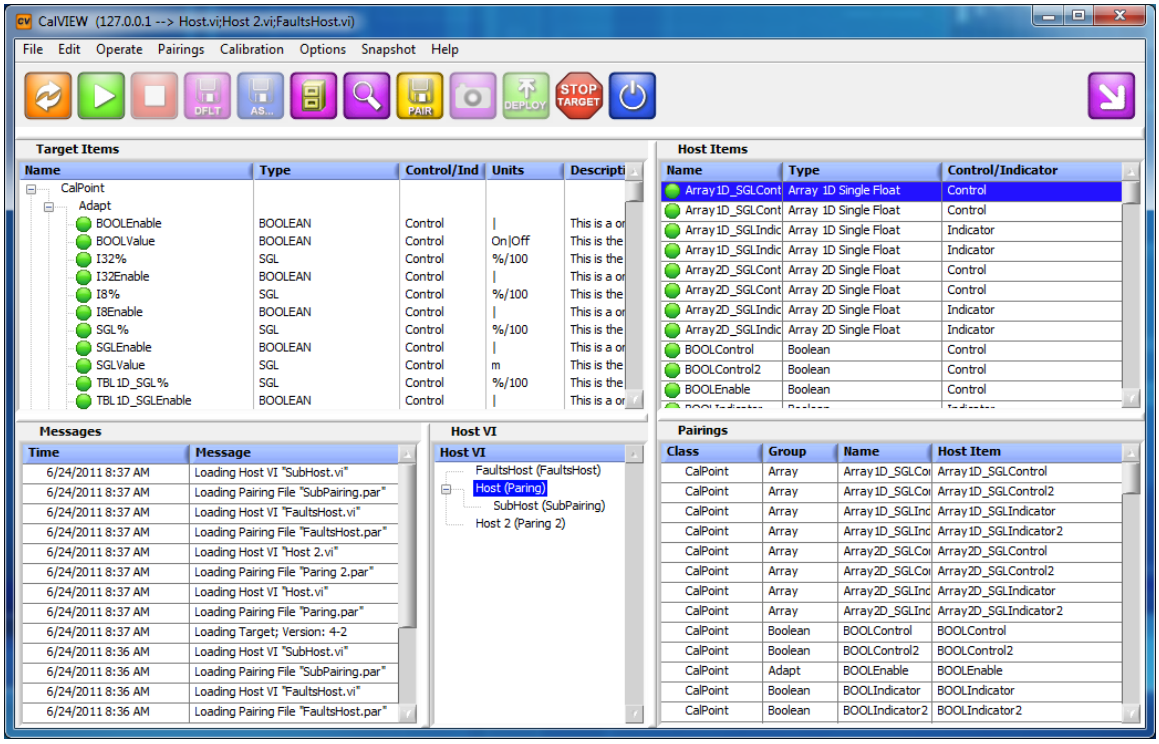

CalVIEW (127.0.0.1 --> Host.vi;Host 2.vi;FaultsHost.vi)
File Edit Operate Pairings Calibration Options Snapshot Help
DFLT
AS...
PAIR
DEPLOY
STOP TARGET
Target Items
Name Type Control/Ind Units Descripti
CalPoint
Adapt
BOOLEnable BOOLEAN Control |
BOOLValue BOOLEAN Control On|Off
I32% SGL Control %/100
I32Enable BOOLEAN Control |
I8% SGL Control %/100
I8Enable BOOLEAN Control |
SGL% SGL Control %/100
SGLEnable BOOLEAN Control |
SGLValue SGL Control m
TBL1D_SGL% SGL Control %/100
TBL1D_SGLEnable BOOLEAN Control |
Host Items
Name Type Control/Indicator
Array1D_SGLCont Array 1D Single Float Control
Array1D_SGLCont Array 1D Single Float Control
Array1D_SGLIndic Array 1D Single Float Indicator
Array1D_SGLIndic Array 1D Single Float Indicator
Array2D_SGLCont Array 2D Single Float Control
Array2D_SGLCont Array 2D Single Float Control
Array2D_SGLIndic Array 2D Single Float Indicator
Array2D_SGLIndic Array 2D Single Float Indicator
BOOLControl Boolean Control
BOOLControl2 Boolean Control
BOOLEnable Boolean Control
Messages
Time Message
6/24/2011 8:37 AM Loading Host VI "SubHost.vi"
6/24/2011 8:37 AM Loading Pairing File "SubPairing.par"
6/24/2011 8:37 AM Loading Host VI "FaultsHost.vi"
6/24/2011 8:37 AM Loading Pairing File "FaultsHost.par"
6/24/2011 8:37 AM Loading Host VI "Host 2.vi"
6/24/2011 8:37 AM Loading Pairing File "Paring 2.par"
6/24/2011 8:37 AM Loading Host VI "Host.vi"
6/24/2011 8:37 AM Loading Pairing File "Paring.par"
6/24/2011 8:37 AM Loading Target; Version: 4-2
6/24/2011 8:36 AM Loading Host VI "SubHost.vi"
6/24/2011 8:36 AM Loading Pairing File "SubPairing.par"
6/24/2011 8:36 AM Loading Host VI "FaultsHost.vi"
6/24/2011 8:36 AM Loading Pairing File "FaultsHost.par"
Host VI
Host VI
FaultsHost (FaultsHost)
Host (Paring)
SubHost (SubPairing)
Host 2 (Paring 2)
Pairings
Class Group Name Host Item
CalPoint Array Array1D_SGLCo Array1D_SGLControl
CalPoint Array Array1D_SGLCo Array1D_SGLControl2
CalPoint Array Array1D_SGLInc Array1D_SGLIndicator
CalPoint Array Array1D_SGLInc Array1D_SGLIndicator2
CalPoint Array Array2D_SGLCo Array2D_SGLControl
CalPoint Array Array2D_SGLCo Array2D_SGLControl2
CalPoint Array Array2D_SGLInc Array2D_SGLIndicator
CalPoint Array Array2D_SGLInc Array2D_SGLIndicator2
CalPoint Boolean BOOLControl BOOLControl
CalPoint Boolean BOOLControl2 BOOLControl2
CalPoint Adapt BOOLEnable BOOLEnable
CalPoint Boolean BOOLIndicator BOOLIndicator
CalPoint Boolean BOOLIndicator2 BOOLIndicator2

Figure 50: CalVIEW Console

CalVIEW Console Window Panes: The CalVIEW console has four main sub-windows which can be individually sized:

Target Items: This window displays a tree of CalPoints available on the target. The tree structure is organized according to the Class, Group and Name given to each CalPoint. In order to populate this window with CalPoints, you must tell the CalVIEW console where the target exists on the network. Also, an application with CalVIEW correctly implemented must be running on the target. You can right click inside the Target Items window and select “Set New Target Address” from the context menu. A dialog will pop-up and make an attempt to find an RT target on the network. If it finds one it will list it in the drop-down list at the top of the dialog. Otherwise you can type in an IP address of your target. After the target is discovered, the Target Items tree will populate. This target location information is retained automatically the next time the CalVIEW console is started and can be changed to a different target at any time. When the target location has been specified, it can be confirmed by dropping down the File menu at the top of the console. The target name and/or IP address will be listed in parenthesis.

If a target with a running application is not available, select “Target Simulation” from the “Select Target” drop-down list, and a CalPoint file may be loaded instead of specifying an IP address in the “Set New Target Address” dialog. The CalPoint text file is generated by the CalVIEW Master dialog “Export to File” function. This will cause the CalVIEW console to load the Target Items list as if a running target were present.

When communicating with a running target, if any changes are made to the target CalPoints then the CalVIEW console refresh button must be pressed in order to update the list of CalPoints in the Target Items window. Proper CalPoint communication between the target and host depend on a correct list of CalPoints.

Controls and indicators which are successfully paired with target CalPoints will have an adjacent green orb ( ).

Host VI: This window displays the list of top level host and sub-hosts. Selecting a host VI causes the Host Items and Pairings to display the items associated with that item.

Host Items: This window displays a list of all controls and indicators in the host VI. In order to populate this window with controls/indicators, you must tell the CalVIEW console where the host VI exists. You can right click inside the Host Items window and select “Set New Host VI” from the context menu. A file dialog will prompt you to select the host VI. The Host Items list will populate with all controls and indicators within the host VI. The name, data type and direction will be listed for each variable. The host VI path is retained automatically the next time the CalVIEW console is started and can be changed to a different path at any time. When the path has been specified, it can be confirmed by dropping

down the File menu at the top of the console. The host VI path will be listed in parenthesis.

Controls and indicators which are successfully paired with target CalPoints will have an adjacent green orb ( ).

Pairings: This window displays the pairings between the target CalPoints and the host controls/indicators. The window can be populated with an existing pairing file by right clicking inside the window and selecting "Load Pairing File." A file dialog will prompt you for selecting a file having an extension of .par. If a pairing file does not exist, then pairings can be created using the drag-and-drop method or the keyboard shortcuts. Pairing options are available under the Pairings menu which aide the process of making pairings.

Pairing Cursor Icons:

This icon appears when dragging a host item to the target items and the cursor is still over the host items.

This icon appears when dragging a target item to the host items and the cursor is still over the target items.

This icon appears when the cursor is over the appropriate list to make a pairing. The item in the list will be highlighted in green if the match is valid and orange if the match is not valid.

Messages: This window reports messages associated with CalVIEW actions, warnings or errors. Warnings and errors are color coded. Each message is time-stamped and the most recent message is inserted at the top of the list.

Menu Items

- **File:**
  - **Select Target (…):** Opens the Select Target dialog.
  - **Select Hosts (…):** Opens the Select Host VI dialog.
  - **Save Pairings:** Saves the pairing files.
  - **Revert Pairings:** Discards current pairings and reloads the pairing files from disk.
  - **Select Units File (…):** Opens the Select Units File dialog.
  - **Select Workspace (…):** Opens the Select Workspace dialog.
  - **Save Workspace:** Saves the workspace.
  - **Select Target VI (…):** Opens the Select Target VI dialog.
  - **Select Fault Sounds:** Opens the Select Fault Sounds dialog.
  - **Backup Target:** Opens the Real-Time Backup/Restore Dialog.
  - **Load Settings:** Loads the CalVIEW settings file from a file.
  - **Save Settings:** Opens the Saves CalVIEW settings file dialog.
  - **Exit:** Stops communication and exits the program.

- **Edit**
  - **Open Target:** Opens target VI for editing in LabVIEW if available.
  - **Open Host VIs:** Opens host VI for editing in LabVIEW if available.
  - **Edit Units File:** Opens the Edit Units dialog.

- **Operate:**
  - **Mini Mode:** Switches the console to mini mode.
  - **Refresh:** Reloads the Target Items and Host Items lists and checks the pairings for errors.
  - **Start Host Application:** Opens host VI and starts communication.
  - **Stop Host Application:** Stops communication and closes host VI.
  - **Stop Target Application:** Stops the target application.
  - **Deploy Target:** Opens and starts target VI in LabVIEW if available.
  - **Silence Alarms:** Opens the Silence Alarms dialog.
  - **Set Target Time:** Synchronizes target time with host time if available.
  - **Open Communication Statistics:** Opens a Communication Statistics Dialog Box.

- **Pairings:**
  - **Auto Pair Items:** Opens a dialog to automatically pair all CalPoints with similar names.
  - **Remove Bad Pairings:** Removes all invalid pairings from the pairings list.

- **Calibration:**
  - **Calibration Difference:** Opens the calibration differencing utility.
  - **Calibration Manager:** Opens the calibration manager utility.
  - **Save Calibration:** Saves the default calibration file to the target. The filename is specified by the "Calibration File" input to the CalVIEW Master VI and is placed in the default data directory on the target.
  - **Save Calibration As …:** Saves the calibration file to the target. The filename is specified by the user input at the prompt and is placed in the default data directory on the target.
  - **Get Calibration File:** Returns the name of the current calibration file in the message window.
  - **Set Calibration File:** Changes the current calibration file when using calibration file pointer file on the target.
  - **Load Calibration File:** Loads a calibration file while running if Live Load is enabled.

- **Options:**
  - **Options:** Opens the Options dialog.
  - **Set Fault Update Rate:** Opens the Set Fault Update Rate dialog.
  - **Set Communication Time:** Opens the Set Communication Time dialog.

- **Snapshot**:
  - **Output File:** Sets the path to the TDMS file to receive CalPoint snapshots.
  - **Log Spec File:** Sets the file path to the snapshot log specifications file.
  - **Take Snapshot:** Takes a snapshot of the current CalPoint values specified in the log spec file.

- **Help:**
  - **About:** Shows CalVIEW version number.
  - **Show Shortcuts:** Displays shortcut keys found in CalVIEWConfig.ini.
  - **Open Manual:** Opens PDF CalVIEW User's Manual.
  - **Open Website:** Opens the Drivven website in the default web browser.
  - **Activate CalVIEW:** Opens Activate CalVIEW dialog.

CalVIEW Controls

Some controls are enabled and disabled automatically according to context.

Refresh Target Items, Host Items, and Pairings

Starts the host application and begins communication

Stops the host application and stops communication

Saves the current calibration as the default target startup calibration

Saves the current calibration on the target as a user specified filename

Opens the calibration file manager tool (display and set calibration file when running)

Opens the CalDiff tool (opens the current calibration file when running)

Saves the pairing file with a user specified filename

Opens and launches the target VI using LabVIEW

Stops the target application and all communication

Saves a snapshot of specified CalPoints to TDMS format

Stop communications and exit CalVIEW

Changes the CalVIEW console to Mini Mode

Mini Mode

Figure 51: CalVIEW Console Mini Mode

The CalVIEW console mini mode contains small versions of the CalVIEW console controls and the most recent message from the message window. The mini console will hover over the host application allowing easy access to CalVIEW console features. The mini console can be moved by dragging the ✣ icon.

CalVIEWConfig.ini

The CalVIEWConfig.ini file contains user preferences and settings. The file is located in the “c:\Program Files\Drivven\CalVIEW\Resources” directory. The user is not required to set any of the values in the file, but can do so if desired for specific console startup control. All parameters, except keyboard shortcut definitions, are set automatically by the CalVIEW console and will be overwritten at various times while the program runs. The file is divided into several sections.

Paths: Contains the most recently used target address, host VI, pairing file and units file

Name	Default Value	Type
TargetAddress	0.0.0.0	String
HostVI	""	String
ParingFile	""	String
UnitsFile	""	String
SnapshotSpecFile	""	String
SnapshotOutputFile	""	String
TargetVI	""	String
WorkspaceFile	""	String

Options: Contains the most recent state of all menu options

Name	Default Value	Type
AUTO_SAVE_PAIRING	FALSE	Boolean
AUTO_SAVE_WORKSPACE	FALSE	Boolean
COLLAPSE_TARGET_TREE	FALSE	Boolean
DISABLE_TIP_STRIPS	FALSE	Boolean
FILTER_BY_TYPE	FALSE	Boolean
HIDE_ITEMS	FALSE	Boolean
HIDE_PAIRING_ITEMS	FALSE	Boolean
HIGHLIGHT_MATCHING_ITEMS	TRUE	Boolean
SAVE_ONLY_CONTROLS	FALSE	Boolean
CLASSIC_PAIRING_ICON	FALSE	Boolean
SHOW_HOST_MENUS	FALSE	Boolean
CommLoopTime	50	I32
FaultUpdateRate	50	I32

Shortcuts: Defines keyboard shortcut keys

Name	Default Value	Type
TARGET_UP	q,Num-Home	String
TARGET_DOWN	a,Num-Left	String
TARGET_NEXT	z,Num-End	String
HOST_UP	w,Num-Up	String
HOST_DOWN	s,Num-Clear	String
HOST_NEXT	x,Num-Down	String
PAIR_UP	e,Num-PageUp	String
PAIR_DOWN	d,Num-Right	String
PAIR_NEXT	c,Num-PageDown	String
PAIR	Space,Num-Insert	String
UNPAIR	Delete,Num-Delete	String
START_HOST	Shift-r	String
STOP_HOST	Shift-s	String
LOAD_CAL	Ctrl-o	String
SAVE_CAL	Ctrl-w	String
SAVE_CAL_AS_DEFAULT	Ctrl-s	String

Window: Contains window size and state

Name	Default Value	Type
Monitor	1	I32
State	1	I32
Left	0	I32
Top	0	I32
Right	800	I32
Bottom	600	I32
Mini_left	0	I32
Mini_Top	0	I32
Mini_Monitor	1	I32
Splitter Center	300	I32
Splitter Left	300	I32
Splitter Right	300	I32

Faults: Contains fault sound files

Name	Default Value	Type
VeryLow	""	String
Low	""	String
Medium	""	String
High	""	String
VeryHigh	""	String

Saving and Loading CalVIEW settings

A CalVIEW settings file may be created with the utility shown in Figure 52. This utility is opened by selecting "Save CalVIEW Settings" from the File menu. The utility saves a variety of startup configuration settings to a file having an extension of .cvsf. The settings include...

- **IP Address:** Target IP address, use 127.0.0.1 for the local host
- **Host VI:** Full path to the host VI
- **Pairing File:** Full path to the pairing file
- **Project-Target-VI:** Full path to the project , target, and VI
- **Units File:** Full path to the units file
- **Workspace File:** Full path to the workspace file
- **Snapshot Spec File:** Full path to the log specifications file
- **Snapshot Output File:** Full path to the snapshot output file
- **Fault Alarms:** Full path to the fault alarm sound files
- **Communication Time:** Desired host update interval
- **Fault Update Rate:** Desired host update rate for fault points
- **Transparent Mode:** Attempts to run the host without displaying the console
- **Mini Mode:** Starts CalVIEW console in mini mode

- **Auto Start Host:** Starts the host application automatically if there are no target connection errors
- **Download Remote Files:** FTP the host VI, pairing file, and units file from the target if available, only the file name (not the file path) should be specified for the files
- **Start Local Target EXE:** Starts an executable which runs on the local host
- **Show Host Menus:** Adds a menu bar to top level hosts to control CalVIEW Console

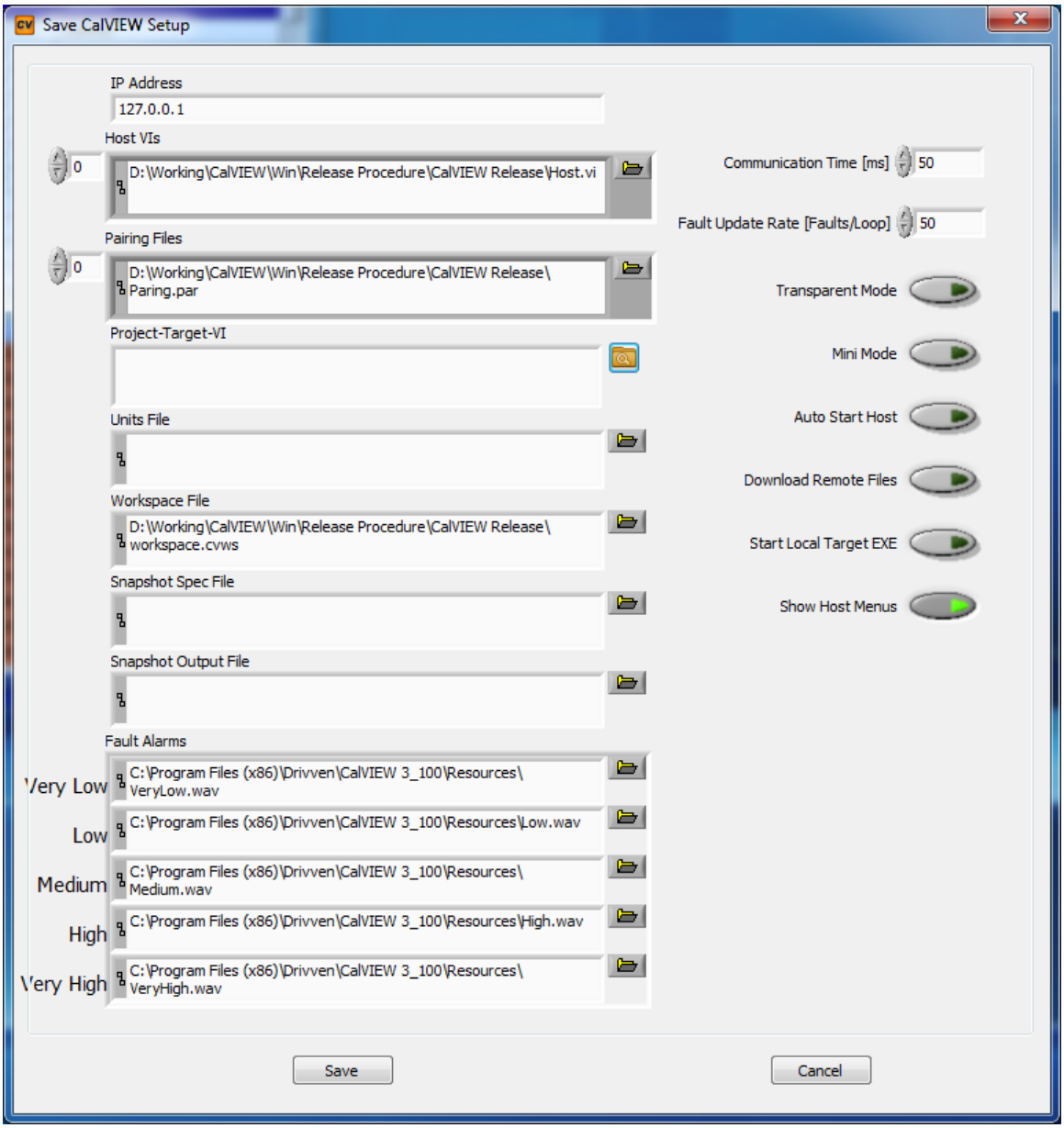


Figure 52: Saving CalVIEW Setting

Some of the parameters stored in the .cvsf are similar to those in the CalVIEWConfig.ini file (TargetAddress, HostVI, etc.). CalVIEWConfig.ini is always referenced when CalVIEW.exe opens and is maintained during CalVIEW operation. However, the .cvsf file may be used as a command line argument to CalVIEW.exe and the settings in the .cvsf file will override similar settings in CalVIEWConfig.ini. This is useful for creating different .cvsf files for different projects so that a CalVIEW environment may be loaded quickly. Furthermore, a .cvsf file will be associated with CalVIEW.exe by the CalVIEW installer. Double clicking a .cvsf file or a shortcut to a .cvsf file will open CalVIEW.exe using the settings within the .cvsf file.

Finding a Target

The user must specify the location of the real-time target to CalVIEW by selecting the “Select Target Machine” option in the File menu. The user may search for real-time targets running CalVIEW on the same local subnet (Figure 53).

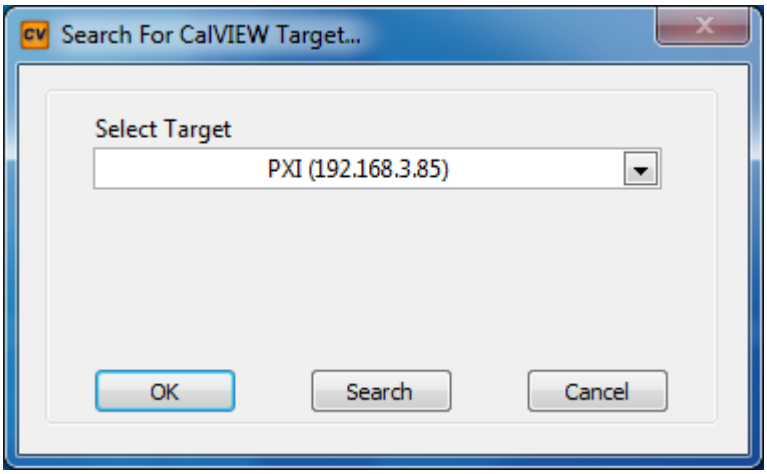


Figure 53: Find Target Dialog Box

The search dialog box will also check the real-time target’s default data directory for .hsf files. If these files are found, they can be downloaded from the target to use on the host (Figure 54). The .hsf file is an ini file with a Host group and three properties. The three properties are Host, Pairings, and Units. The Host and Pairing properties are lists of files paths relative to the default data directory and are separated by a “;”. The Units is a single path relative to the default data directory. With serviceable targets in the field, it is useful to keep the host VI, pairing file and units file with the target so that an operator going out to service only needs the CalVIEW console application. The target must not be password protected. If there appears to be an acceptable set of files on the target, you will be prompted to save the files to local disk by pressing the Download Files button, followed by pressing the OK button. The files will then be downloaded to your default data directory on your computer. After the files are downloaded, a console connection will be made to the specified target, and the downloaded host VI and pairing file will be loaded into the console, ready for starting communication.

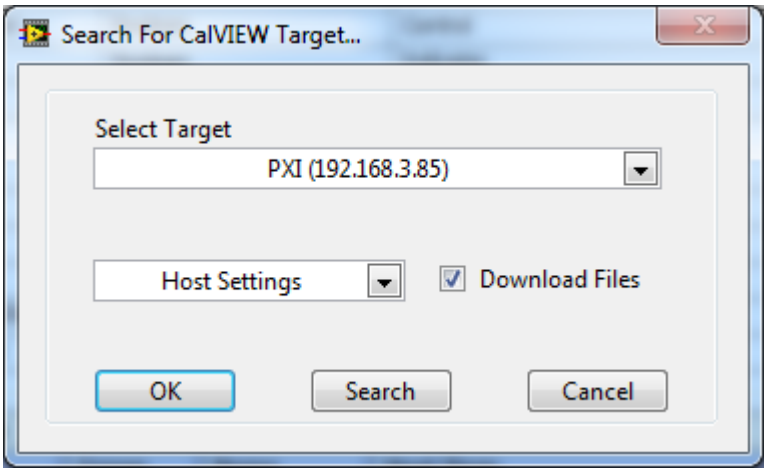


Figure 54: Download Host Files

The .hsf file is an ini file with a Host group and three properties. The three properties are Host, Pairings, and Units. The Host and Pairing properties are lists of files paths relative to the default data directory and are separated by a ";". The Units is a single path relative to the default data directory.

Host: Contains the paths to the host Vis, pairing files, and units file

Property	Type
Host	String
Pairing	String
Units	String

Example:

```
[Host]
Host = "Host_1.vi;Host_2.vi"
Pairing = "Pairing_1.par;Pairing_2.par"
Units = ""
```

The user may also manually enter an IP address (Figure 55) or select a file exported by the CalVIEW Master to simulate a target running an application with CalVIEW implemented (Figure 56). A large red indicator will appear next to the console Stop button when CalVIEW is in simulation mode.

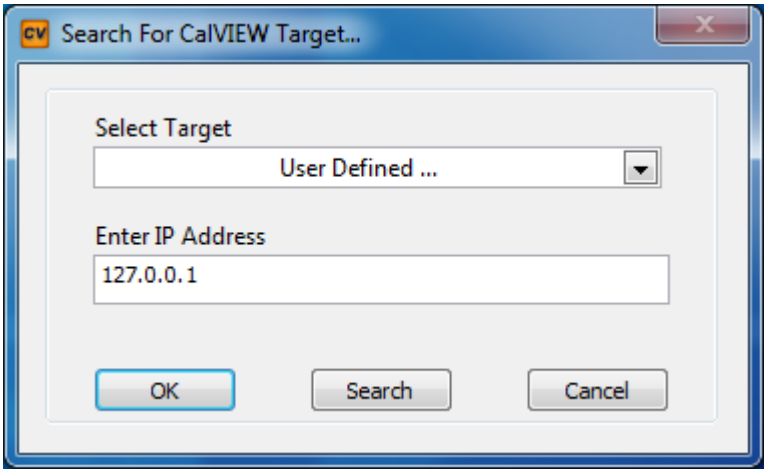


Figure 55: User Defined IP Address

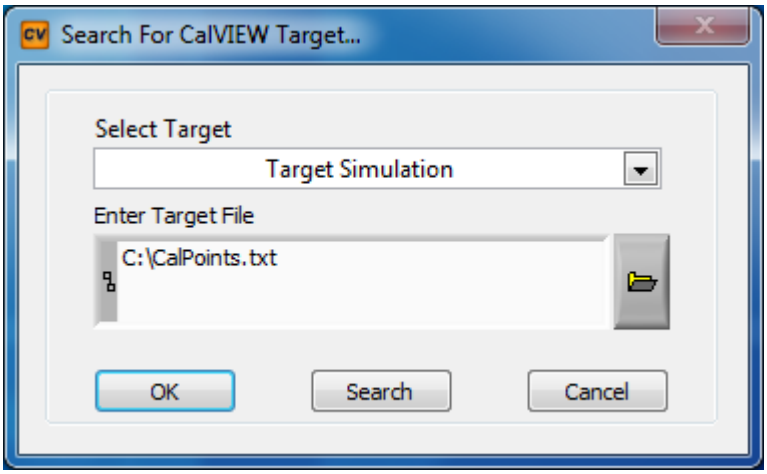


Figure 56: Target Simulation

Pairing CalPoints

After the user has selected a host VI and a target, the CalVIEW console program allows the user to pair host controls and indicators to the CalPoints on the target. The user can drag items from the Target Items list to the Host Items list as well as the reverse. The items can also be paired using the shortcut keys defined by the CalVIEWConfig.ini file. Errors will cause the Paired Items list to turn red and an X will be displayed by the item with an error. The message display area shows status messages. Problems reading files or connecting to the target will be displayed with orange highlight. The pairing must be made while the host VI is not running (communication is stopped).

Taking a Snapshot

CalVIEW has the ability to take a snapshot of the current values of specified host items (excluding tables and arrays). The snapshot is stored in a TDMS file which is specified by the user. The data included in the snapshot is defined by the log specifications file. The snapshot spec file is in the same format as the pairing file with the addition of a 5th column. The pairing file is a good starting point for creating a snapshot spec file. The 5th column defines the TDMS group which the data will be stored in. If no group is specified, the default group will be used. The same value may be placed in multiple groups but should not be placed in a group twice. If the same CalPoint is placed in the same group more than once, each snapshot will use more than one TDMS line. The name of the CalPoint in the TDMS file will show up as the host name, using the host units. If a new CalPoint is introduced to a pre-existing TDMS output file, the TDMS file will be back-filled with zeroes (for the new CalPoint) by CalVIEW.

CalDiff (Calibration File Difference)

CalDiff is a calibration file differencing utility that allows a user to easily compare and modify calibration files (Figure 57). The utility opens two files specified at the top of each side of the tool. To load a file, enter the path or use the browse button to find the file. The list of calibrations automatically updates when the path changes. The user may double click an item to push it to the other file. The user may also select one or more items and use the right click menu options to manipulate the data.

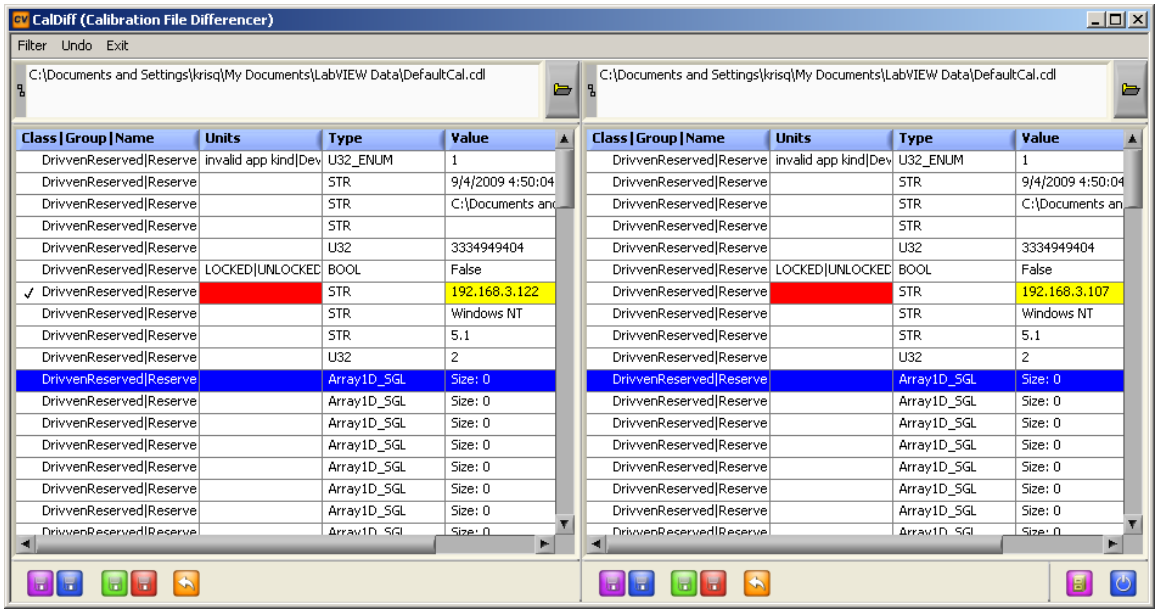


Figure 57: Calibration File Difference

- **Filter:**
  - **Differences Only:** Only shows differences in the file
  - **Type:** Shows only the calibrations for a specified data type
  - **Clear All:** Clears all filter settings
- **Undo:** Undo the last change, up to 10 changes
- **Exit:** Closes the differencing tool

Saves the current data to the loaded filename

Saves the current data to a new file

Exports the current data to a TDMS file

Exports the current data to a CSV file

Reloads the current file

Opens the Calibration File Manager (displays and sets viewed files)

Closes the calibration differencing tool

Calibration File Manager

The calibration file manager is used to copy files between the target and host machines. The file manager starts from the CalVIEW console or the CalDiff. It will operate in a different way depending on where it was started.

When running the file manager from the CalVIEW console, only the current target may be viewed. When communicating with a target, the user is allowed to view and set the default calibration file.

When running the file manager from the CalDiff, the user may view and set the files opened on the left and right side of the tool.

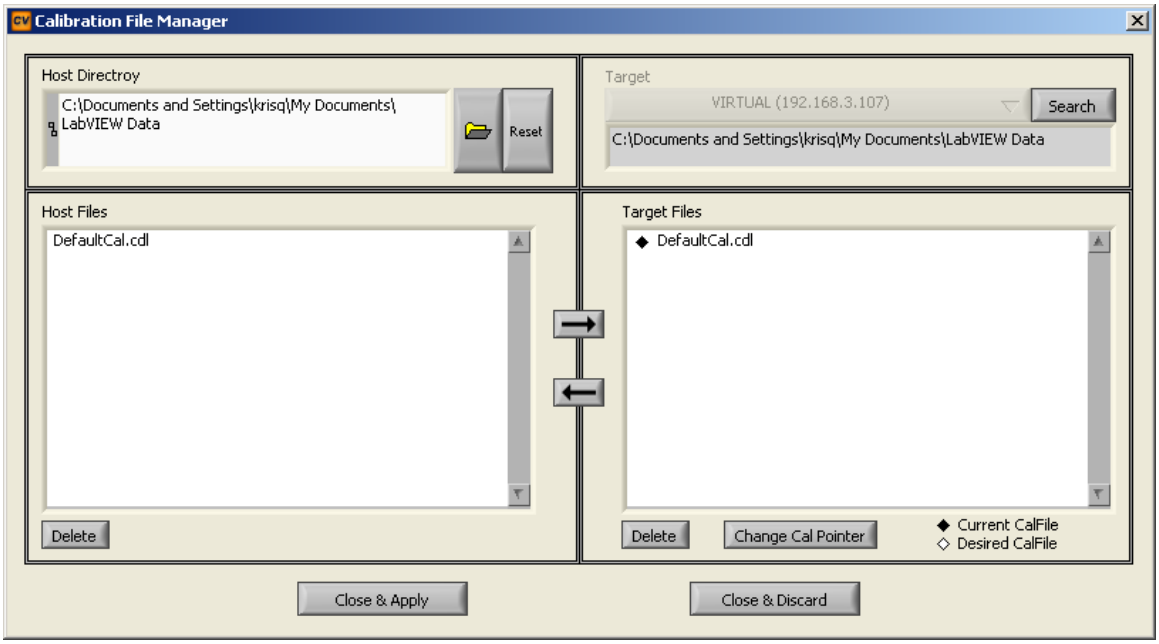


Figure 58: Calibration File Manager

A host computer directory should be specified. This is the location where files will be copied to and from the host. The default directory is the “LabVIEW Data” directory. Only .cdl calibration files are displayed.

The target list is populated with all targets running CalVIEW. Selecting a target displays all the calibration files found in the “LabVIEW Data” directory of the target.

Files may be copied using the set of arrows. During file transfer, the file name may be modified. Files may be deleted from the target or host using the delete button.

When opened from the CalDiff, the “Host Files” list indicates the currently loaded files. It may also be used to modify which files are opened.

When opened from the CalVIEW console while communicating with a target, the “Target File” indicates the current calibration file on the target. It may also be used to change the current calibration file.

Unit Conversions

Each CalPoint provides a field for specifying the engineering units of the parameter. In order to display the units along with the graphical controls or indicators on the host VI, the control's caption must be made visible in the editable version of the host VI with a minimal amount of text. The actual text within the caption is not important because CalVIEW will fill in the caption with the text provided with the CalPoint units field. The CalVIEW console also provides a unit conversion utility so that the host VI can display different units than what are used on the target. The unit conversion utility allows the user to build a units conversion file which provide conversion factors. CalVIEW will convert the values to the requested units as they are transmitted to and from the target. The unit conversions are defined in the Unit Editor dialog as shown in Figure 59. CalVIEW automatically converts the units based on the operator's preferences. The units file has a .unt extension and is associated with a file of the same name having a .duf extension describing the default units. Drivven provides a sample units file found in the CalVIEW resources folder located at "c:\Program Files\Drivven\CalVIEW\Resources\Units.unt". The Unit Editor summarizes all of the unit conversions in the units file and allows the user to open three different dialogs for editing unit classes, unit conversions, and default units. A unit class describes a collection of associated unit conversions such as Pressure or Length. Complex units such as Speed are a combination of base units and must have their own class. CalVIEW will not combine units from different classes such as Length/Time. The "Save File" button commits the current configuration to the units file. The OK button saves the file and exits, whereas the Cancel button exits the unit editor without saving the file.

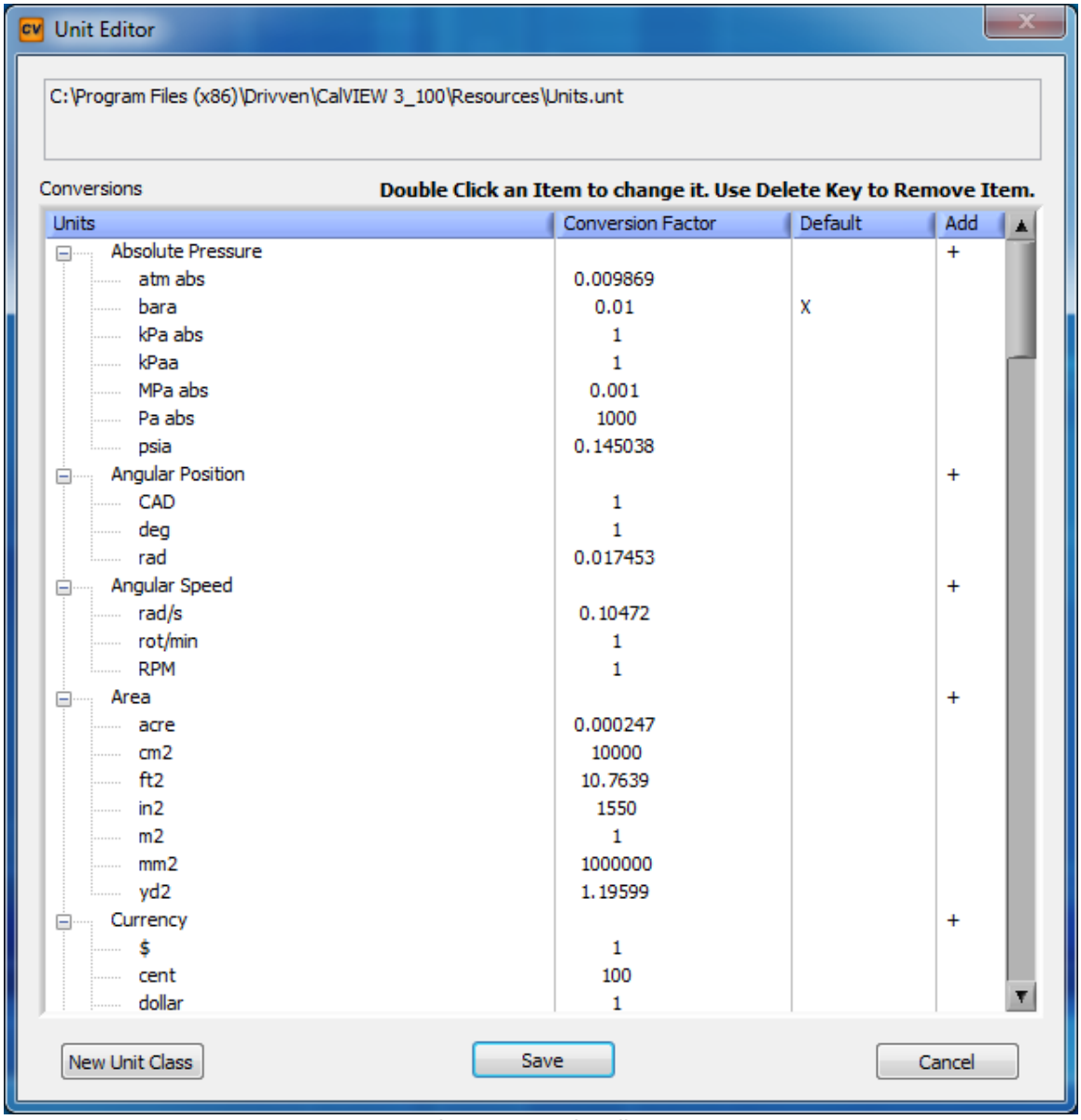


Figure 59: Unit Editor

The unit editor allows the user to add/remove unit conversions, add/remove unit classes, modify the unit conversions, and change the default display unit. Double clicking the first 2 columns opens a dialog box to change the conversion factor of a unit conversion (Figure 60) or edit the unit class name. Double clicking on the default column changes the default unit of a unit class. The delete key removes the selected unit conversion or class. Double clicking the + in the add column adds a unit conversion to the unit class. The “New Unit Class” button adds a unit class. Each unit class and each unit conversion must be unique. The conversion factors must be non-zero values. Errors are displayed with a red background.

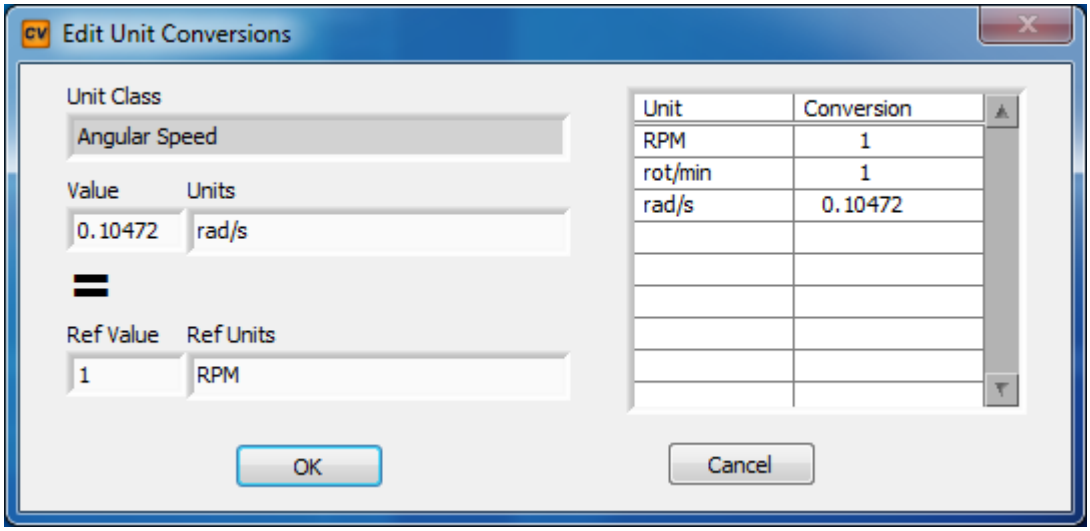


Figure 60: Edit Unit Conversion

Unit Conversion Rules

Unit conversions follow some simple rules.

- If host VI controls/indicators caption fields contain brackets, braces, or parentheses, then the units are enclosed in brackets, braces, or parentheses.
- If host VI controls/indicators caption fields contain “@”, then conversions are ignored.
- If host VI controls/indicators caption fields contain “#”, then CalVIEW converts to units specified after “#” if the conversion exists. The default units will not be used.
- Editable host VI must make caption visible with at least one character.
- Unit file must be set and the Enable Unit Conversions must be checked under Options menu.

Example of Unit Conversions

- Assumed Target Units: “psi”
- Assumed Default Unit: “kPa”
- … refers to any character

Editable Host VI Caption Contains…	CalVIEW Host VI Displays…	Comments
…	kPa	Adds units
{…}	{kPa}	Leaves braces
[…]	[kPa]	Leaves brackets
(…)	(kPa)	Leaves parentheses
@...	@psi	Use target units (ignore default units)
#Pa	#Pa	Use requested host units if conversion exists
#ft	kPa	Use default host units if no conversion exists
[@atm]	[@psi]	Combination
{#atm}	{#atm}	Combination
psi|in|sec	kPa|in|sec	Multiple Units (Tables)

Simple Example

Basic Project Example

A LabVIEW project which implements CalVIEW should contain a real-time target with a real-time VI as shown in Figure 61. The following example project is included in the Installer and can be found in the LabVIEW example finder.

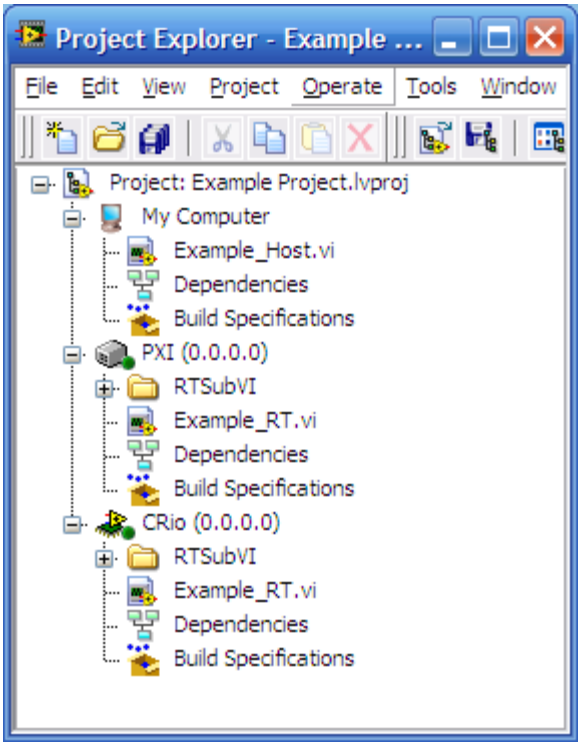


Figure 61: Example Project

The top level RT VI should contain four main parts as shown in Figure 62.

1. **CalVIEW Master VI:** Loads calibration file and initializes CalVIEW
2. **Case Structure:** Prevents loops from running if initialization error occurs.
3. **Loop(s) Containing CalPoint(s):** CalPoints may exist in SubVIs. SubVIs with CalPoints must be added to the project under the real-time target. Creating a project folder such as “RTSubVI” for collecting subVIs is recommended.

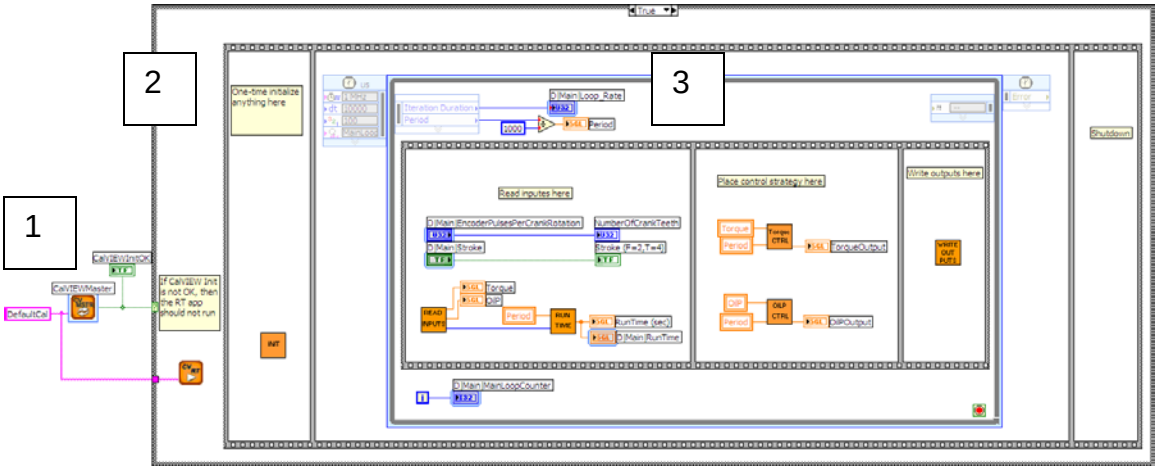


Figure 62: Basic CalVIEW Project Example RT VI

Frequently Asked Questions

Why are some CalPoints not found by CalVIEW Master?

- All VIs containing CalPoints must be included in the project under the target.
- CalPoints in disable diagram structures will not be found by CalVIEW Master.

Why are CalPoints behaving unexpectedly?

- CalVIEW Master VI Configuration Dialog must be opened before deploying in order to build CalPoint table for run-time.
- The target VI must be started from a LabVIEW project.
- Limits and slew rates for U32 and I32 CalPoints are represented internally as float32. Because of this not all 32 bit numbers are available, especially those that cannot be represented in less than 24 bits.

What are some restrictions for CalPoint Name?

- A "." (period) is not allowed in any part of a CalPoint name.
- A "|" (pipe) is used as the separator for CalPoint Class|Group|Name and units. It should not be used within these individual fields of the configuration dialog.

Why is LabVIEW unable to find some VIs while opening a CalVIEW target VI?

- Open and close the CalVIEW Master VI to update it to the latest version.
- When opening a VI under the "My Computer" target, some real time libraries are unavailable and should be ignored ("RT Set Date and Time.vi"). Ignoring the VIs should not cause errors.
- CalVIEW installs to the latest version of LabVIEW installed on a computer. If you are running an older version of LabVIEW that computer, CalVIEW may not be able to find all the needed files.

Why does deploying CalVIEW project to a cRIO fail?

- CalVIEW_C.out file must be copied to all VxWorks-based targets before deploying RT VIs. Please refer to the section titled "VxWorks Setup".
- "Control CalVIEW" subVI should be placed outside all loops.

What are the requirements for a host VI?

- Host VI must contain a while loop and a wait.

- Host control/indicator names cannot contain carriage return.
- Host control/indicator names must be unique.
- An enumerated item on the host VI must have the same number of enumerated items as the CalPoint. The string for each enumerated item on the host does not matter because the Text Labels from the target CalPoint will be copied over.

Why does the Host VI open but does not run?

- The broken run arrow will show a list of errors. If subVIs are missing, locate them so that the host VI can run.
- Some errors require opening the Host VI in the full version of LabVIEW to fix the error. The broken arrow in the full version of LabVIEW will give details about the errors.

Why am I having trouble pairing two items?

- The CalVIEW console will only allow pairings for target and host items having the same data type and same direction (control or indicator).
- Host Boolean controls must be set to switching mechanical action.
- Host control/indicator names cannot contain carriage return.
- Host control/indicator names must be unique.

Why does the CalVIEW Console not find my target?

- CalVIEW Console will only find targets running a CalVIEW-based project.
- CalVIEW Console can only find targets on the same subnet as the host computer.
- TCP/IP and UDP ports may need to be unblocked in hardware and software firewalls to allow CalVIEW to find targets or transfer files. The list of used ports is listed above.
- CalVIEW may not find the host VI and pairing files stored on the target due to some networks or host computers with security software installed.
- CalVIEW may not connect to the target if the target application was not started from a LabVIEW project. After opening the target VI from the project, run the CalVIEW Master.

Why does the CalVIEW Console not find my Host VI?

- Check the file path to the VI.
- Check the version of the VI. If the VI was last saved using an incompatible version of LabVIEW, open it in the compatible version of LabVIEW and try again. Save for previous version will not work.

- CalVIEW may not load a host VI if it contains a Waveform Graph with certain types of inputs such as clusters. Instead, wire an array directly to the Waveform Graph.

Why is the host and target time off by X hours after setting the target time?

- The target and host times are set using the universal time stamp. If the target time zone is different than the host time zone, the two will have local times that are off by the appropriate number of hours. Set the target time zone using the measurements and automation explorer.

What can I do to run CalVIEW over a slow connection?

- CalScopes and large arrays take large amounts of bandwidth. Limit their use.
- Set the communication loop time (via the CalVIEW Console) to a slower speed.

How can I revert a project to a previous version of CalVIEW?

- CalVIEW 2.86 projects cannot be reverted to earlier versions automatically. CalPoints need to be deleted and recreated using a previously version of CalVIEW.

Online Tools

The CalVIEW product description and pricing may be found at:

- http://www.drivven.com/CalVIEW.htm

The latest version of CalVIEW may be downloaded from:

- http://www.drivven.com/visitor_download/Software/CalVIEW3.100.zip

The latest version of CalVIEW BETA release may be downloaded from:

- http://www.drivven.com/visitor_download/Software/CalVIEW3.110.b.zip

The latest version of this manual may be downloaded from:

- http://www.drivven.com/visitor_download/Manuals/CalVIEW%20User's%20Manual.pdf

An interactive CalVIEW tutorial may be found at:

- http://www.drivven.com/tutorial/CalVIEW/CalVIEW%20Tutorial.html

The LabVIEW 2010 SP1 runtime engine may be downloaded from:

- http://joule.ni.com/nidu/cds/view/p/id/2534/lang/en

The Drivven support web page may be found at:

- http://www.drivven.com/Support.htm

The Drivven contact information may be found at:

- http://www.drivven.com/ContactUs.htm